



US006639524B2

(12) **United States Patent**  
**Schouhamer Immink**

(10) **Patent No.:** **US 6,639,524 B2**  
(45) **Date of Patent:** **Oct. 28, 2003**

(54) **METHOD AND APPARATUS FOR CODING INFORMATION, METHOD AND APPARATUS FOR DECODING CODED INFORMATION, METHOD OF FABRICATING A RECORDING MEDIUM, THE RECORDING MEDIUM AND MODULATED SIGNAL**

(75) Inventor: **Kees A. Schouhamer Immink**, Geldrop (NL)

(73) Assignee: **LG Electronics Inc.**, Seoul (KR)

(\* ) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 321 days.

(21) Appl. No.: **09/813,969**

(22) Filed: **Mar. 22, 2001**

(65) **Prior Publication Data**

US 2001/0030933 A1 Oct. 18, 2001

(30) **Foreign Application Priority Data**

Mar. 22, 2000 (EP) ..... 00201052

(51) **Int. Cl.**<sup>7</sup> ..... **H03M 7/00**; G06F 13/12; G11B 20/00

(52) **U.S. Cl.** ..... **341/51**; 341/59; 341/106; 710/66; 710/68

(58) **Field of Search** ..... 710/65, 66, 70, 710/68; 360/39, 55; 341/95, 58, 52, 59, 106; 714/758, 755

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,115,768 A \* 9/1978 Eggenberger et al.  
4,410,877 A \* 10/1983 Carrasso et al.  
4,517,552 A \* 5/1985 Shirota et al.  
5,696,505 A \* 12/1997 Schouhamer et al.  
6,084,356 A \* 7/2000 Arts

FOREIGN PATENT DOCUMENTS

EP 0745254 B1 \* 4/1998 ..... G11B/20/00

\* cited by examiner

*Primary Examiner*—Kim Huynh

(74) *Attorney, Agent, or Firm*—Birch, Stewart, Kolasch & Birch, LLP

(57) **ABSTRACT**

A coding device and method in which m-bit information words are converted into n-bit code words such that the coding rate m/n is greater than 1/2. The coding device and method are also employed to record information on a recording medium and to transmit information. In the decoding method and apparatus, n-bit code words are decoded into m-bit information words. The decoding involves determining the state of a next n-bit code word, and based on the state determination, the current n-bit code word is converted into an m-bit information word. The decoding device and method are employed to reproduce information from a recording medium, and to receive information transmitted over a medium.

**53 Claims, 18 Drawing Sheets**

SUBGROUP	1st KIND				2nd KIND			3rd KIND	
	STATE 1	STATE 2	STATE 3	STATE 4	STATE 5	STATE 6	STATE 7	STATE 8	STATE 9
E0000	6	4	4	4	0	0	0	0	0
E0010	1	2	2	4	0	0	0	0	0
E0001	1	4	4	1	0	0	1	0	0
E1000	0	0	0	0	4	4	5	0	0
E1010	0	0	0	0	2	2	2	0	0
E1001	0	0	0	0	4	4	1	0	0
E0100	0	0	0	0	0	0	0	4	5
E0110	0	0	0	0	0	0	0	2	2
E0101	0	0	0	0	0	0	0	4	2

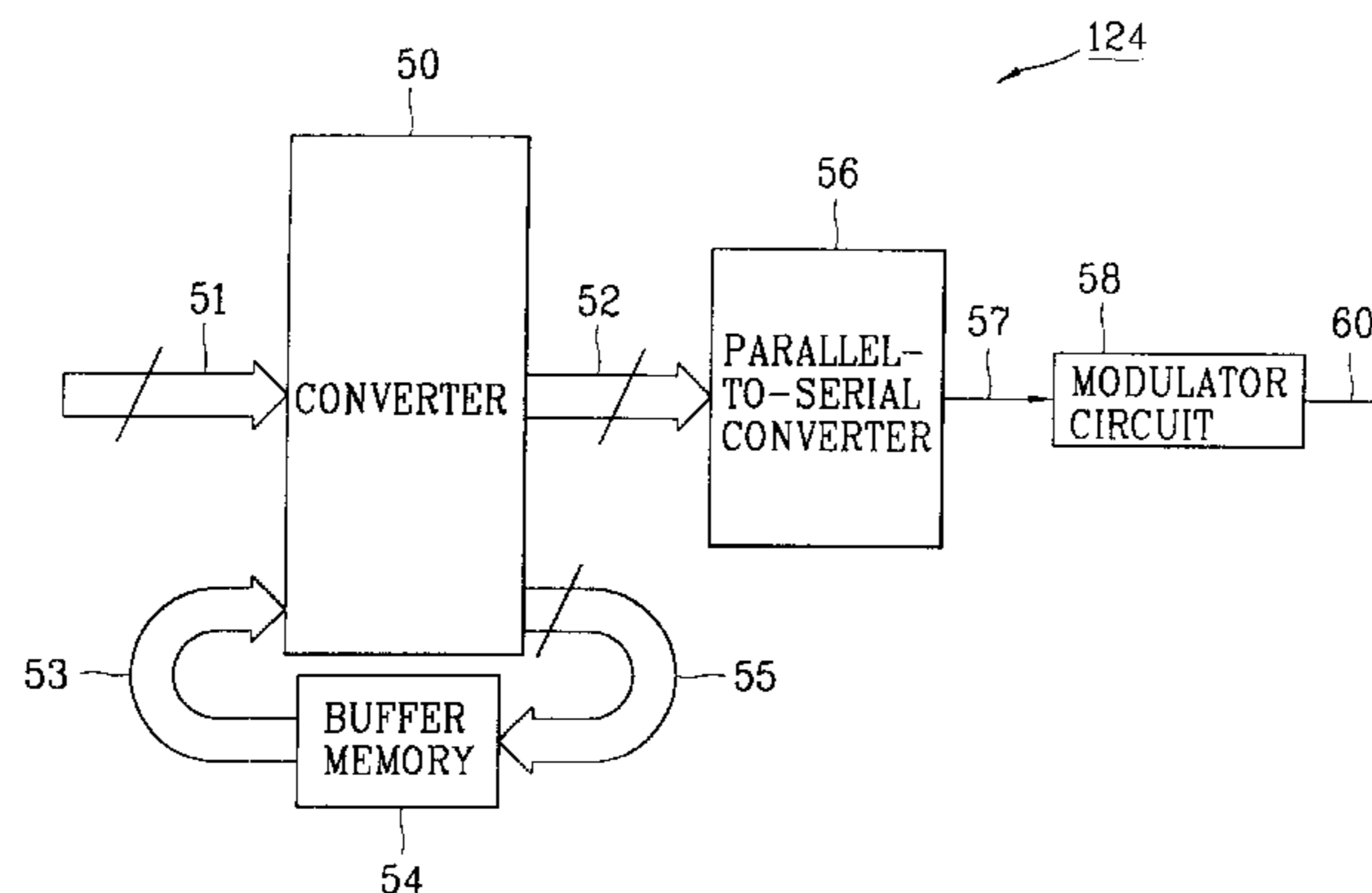




FIG. 3

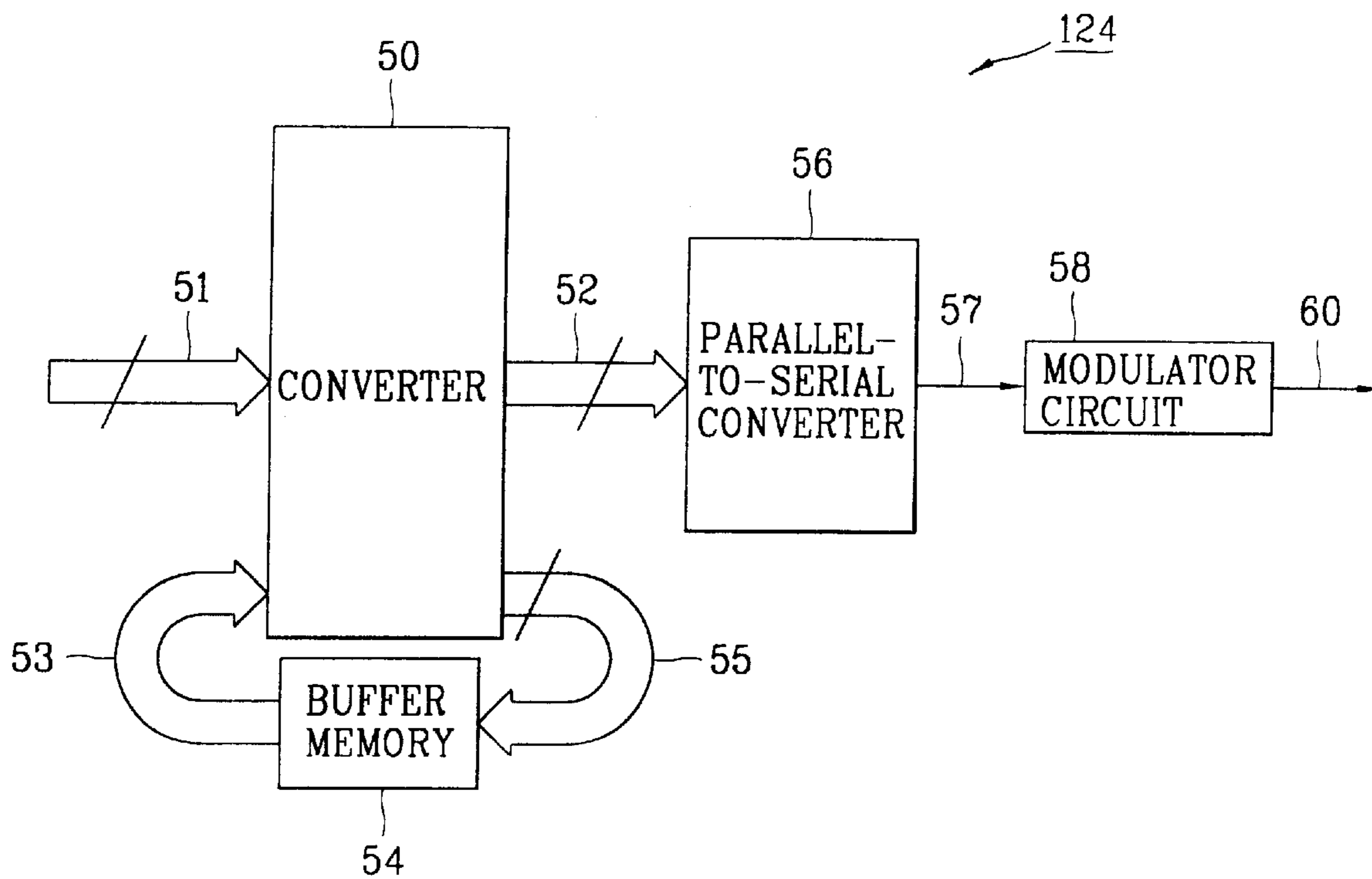


FIG. 4A

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
0	00000000100	10001000100	100010001000	100100001000	10001001000	10000100000	10010010001	40100010000	101001001001
1	00000000100	20001000100	200010001000	200100001000	20001001000	20000000000	20000000000	20100010000	201000000000
2	00000000100	30001000100	300010001000	300100001000	30001001000	30000000000	30000000000	30100010000	301000000000
3	00000000100	40001000100	400010001000	400100001000	40001001000	40000000000	40000000000	40100010000	401000000000
4	00000000100	50001000100	500010001000	500100001000	50001001000	50000000000	50000000000	50100010000	501000000000
5	00000000100	60001000100	600010001000	600100001000	60001001000	60000000000	60000000000	60100010000	601000000000
6	00000000100	70001000100	700010001000	700100001000	70001001000	70000000000	70000000000	70100010000	701000000000
7	00000000100	80001000100	800010001000	800100001000	80001001000	80000000000	80000000000	80100010000	801000000000
8	00000000100	90001000100	900010001000	900100001000	90001001000	90000000000	90000000000	90100010000	901000000000
9	00000000100	10001000100	100010001000	100100001000	10001001000	10000000000	10010000000	10100000000	101000100100
10	00000000100	20001000100	200010001000	200100001000	20001001000	20000000000	20010000000	20100000000	201000100100
11	00000000100	30001000100	300010001000	300100001000	30001001000	30000000000	30010000000	30100000000	301000100100
12	00000000100	40001000100	400010001000	400100001000	40001001000	40000000000	40010000000	40100000000	401000100100
13	00000000100	50001000100	500010001000	500100001000	50001001000	50000000000	50010000000	50100000000	501000100100
14	00000000100	60001000100	600010001000	600100001000	60001001000	60000000000	60010000000	60100000000	601000100100
15	00000000100	70001000100	700010001000	700100001000	70001001000	70000000000	70010000000	70100000000	701000100100
16	00000000100	80001000100	800010001000	800100001000	80001001000	80000000000	80010000000	80100000000	801000100100
17	00000000100	90001000100	900010001000	900100001000	90001001000	90000000000	90010000000	90100000000	901000100100
18	00000010000	100010000000	100010000000	100100000000	100010000000	10000000000	10010000000	10100000000	101001000000
19	00000010000	200010000000	200010000000	200100000000	200010000000	20000000000	20010000000	20100000000	201001000000
20	00000010000	300010000000	300010000000	300100000000	300010000000	30000000000	30010000000	30100000000	301001000000
21	00000010000	400010000000	400010000000	400100000000	400010000000	40000000000	40010000000	40100000000	401001000000
22	00000010000	500010000000	500010000000	500100000000	500010000000	50000000000	50010000000	50100000000	501001000000
23	00000010000	600010000000	600010000000	600100000000	600010000000	60000000000	60010000000	60100000000	601001000000
24	00000010000	700010000000	700010000000	700100000000	700010000000	70000000000	70010000000	70100000000	701001000000
25	00000010000	800010000000	800010000000	800100000000	800010000000	80000000000	80010000000	80100000000	801001000000
26	00000010000	900010000000	900010000000	900100000000	900010000000	90000000000	90010000000	90100000000	901001000000
27	00000010000	100010000000	100010000000	100100000000	100010000000	10000000000	10010000000	10100000000	101001000000
28	00000010000	200010000000	200010000000	200100000000	200010000000	20000000000	20010000000	20100000000	201001000000
29	00000010000	300010000000	300010000000	300100000000	300010000000	30000000000	30010000000	30100000000	301001000000
30	00000010000	400010000000	400010000000	400100000000	400010000000	40000000000	40010000000	40100000000	401001000000
31	00000010000	500010000000	500010000000	500100000000	500010000000	50000000000	50010000000	50100000000	501001000000

FIG. 4B

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9	
Data	Channel bits	state	Channel bits	state	Channel bits	state	Channel bits	state	Channel bits	state
32	0000100000	6 00010000100	6 00100000100	6 00100100100	6 10000010000	6 10001000100	6 10010001000	6 01000010000	6 01001000100	6 01001000100
33	00000100000	7 000100000100	7 001000000100	7 001001000100	7 10000010000	7 100010000100	7 10010001000	7 01000010000	7 010010000100	7 010010000100
34	00000100000	8 000100000100	8 001000000100	8 001001000100	8 10000010000	8 100010000100	8 10010001000	8 01000010000	8 010010000100	8 010010000100
35	00000100000	9 000100000100	9 001000000100	9 001001000100	9 10000010000	9 100010000100	9 10010001000	9 01000010000	9 010010000100	9 010010000100
36	00000100100	1 00000010010	1 00001000010	1 00010010010	1 10000000010	1 10000100010	1 10010010000	1 01000000010	1 01000000010	1 01001001000
37	00000100100	2 00000010010	2 00001000010	2 00010010010	2 10000000010	2 10000100010	2 10010010000	2 01000000010	2 01000000010	2 01001001000
38	00000100100	3 00000010010	3 00001000010	3 00010010010	3 10000000010	3 10000100010	3 10010010000	3 01000000010	3 01000000010	3 01001001000
39	00000100100	4 00000010010	4 00001000010	4 00010010010	4 10000000010	4 10000100010	4 10010010000	4 01000000010	4 01000000010	4 01001001000
40	00000100100	5 00000010010	5 00001000010	5 00010010010	5 10000000010	5 10000100010	5 10010010000	5 01000000010	5 01000000010	5 01001001000
41	00000100100	6 00000010010	6 00001000010	6 00010010010	6 10000000010	6 10000100010	6 10010010000	6 01000000010	6 01000000010	6 01001001000
42	00000100100	7 00000010010	7 00001000010	7 00010010010	7 10000000010	7 10000100010	7 10010010000	7 01000010010	7 01000010010	7 01001001000
43	00000100100	8 00000010010	8 00001000010	8 00010010010	8 10000000010	8 10000100010	8 10010010000	8 01000010010	8 01000010010	8 01001001000
44	00000100100	9 00000010010	9 00001000010	9 00010010010	9 10000000010	9 10000100010	9 10010010000	9 01000010010	9 01000010010	9 01001001000
45	00001000000	1 00000100010	1 00001000010	1 00010000010	1 10000000010	1 10000100010	1 10010000010	1 01000010010	1 01000010010	1 01000100010
46	00001000000	2 00000100010	2 00001000010	2 00010000010	2 10000000010	2 10000100010	2 10010000010	2 01000010010	2 01000010010	2 01000100010
47	00001000000	3 00000100010	3 00001000010	3 00010000010	3 10000000010	3 10000100010	3 10010000010	3 01000010010	3 01000010010	3 01000100010
48	00001000000	4 00000010001	4 00001000001	4 00010000001	4 10000000001	4 10000100001	4 10010000001	4 01000000001	4 01000000001	4 01000100001
49	00001000000	5 00000010001	5 00001000001	5 00010000001	5 10000000001	5 10000100001	5 10010000001	5 01000000001	5 01000000001	5 01000100001
50	00001000000	6 00000010001	6 00001000001	6 00010000001	6 10000000001	6 10000100001	6 10010000001	6 01000000001	6 01000000001	6 01000100001
51	00001000000	7 00000010001	7 00001000001	7 00010000001	7 10000000001	7 10000100001	7 10010000001	7 01000000001	7 01000000001	7 01000100001
52	00001000000	8 00000010001	8 00001000001	8 00010000001	8 10000000001	8 10000100001	8 10010000001	8 01000000001	8 01000000001	8 01000100001
53	00001000000	9 00000010001	9 00001000001	9 00010000001	9 10000000001	9 10000100001	9 10010000001	9 01000000001	9 01000000001	9 01000100001
54	00000000010	1 00000100001	1 00001000001	1 00010010001	1 10000001001	1 10000100001	1 10010010001	1 01000001001	1 01000001001	1 01000100001
55	00000000010	2 00000100001	2 00001000001	2 00010010001	2 10000001001	2 10000100001	2 10010010001	2 01000001001	2 01000001001	2 01000100001
56	00000000010	3 00000100001	3 00001000001	3 00010010001	3 10000001001	3 10000100001	3 10010010001	3 01000001001	3 01000001001	3 01000100001
57	00000000010	4 00000100001	4 00001000001	4 00010010001	4 10000001001	4 10000100001	4 10010010001	4 01000001001	4 01000001001	4 01000100001
58	00000000010	8 00000100001	8 00001000001	8 00010010001	8 10000001001	8 10000100001	8 10010010001	8 01000001001	8 01000001001	8 01000100001
59	00000000010	9 00000100001	9 00001000001	9 00010010001	9 10000001001	9 10000100001	9 10010010001	9 01000001001	9 01000001001	9 01000100001
60	00000001001	1 00000100101	1 00001000001	1 00010000101	1 10000010001	1 10000100001	1 10010000101	1 01000001001	1 01000001001	1 01000100001
61	00000001001	2 00000100101	2 00001000001	2 00010000101	2 10000010001	2 10000100001	2 10010000101	2 01000001001	2 01000001001	2 01000100001
62	00000001001	3 00000100101	3 00001000001	3 00010000101	3 10000010001	3 10000100001	3 10010000101	3 01000001001	3 01000001001	3 01000100001
63	00000001001	4 00000100101	4 00001000001	4 00010000101	4 10000010001	4 10000100001	4 10010000101	4 01000001001	4 01000001001	4 01000100001

FIG. 5

INFORMATION WORDS (DECIMAL NOTATION)	STATE	CODE WORDS	NEXT STATE
000001 (1)	S1	00000000100	S2
000011 (3)	S2	00001000100	S4
000101 (5)	S4	00100001000	S6
001100 (12)	S6	10000100100	S4
010011 (19)	S4	00100100000	S2

FIG. 6

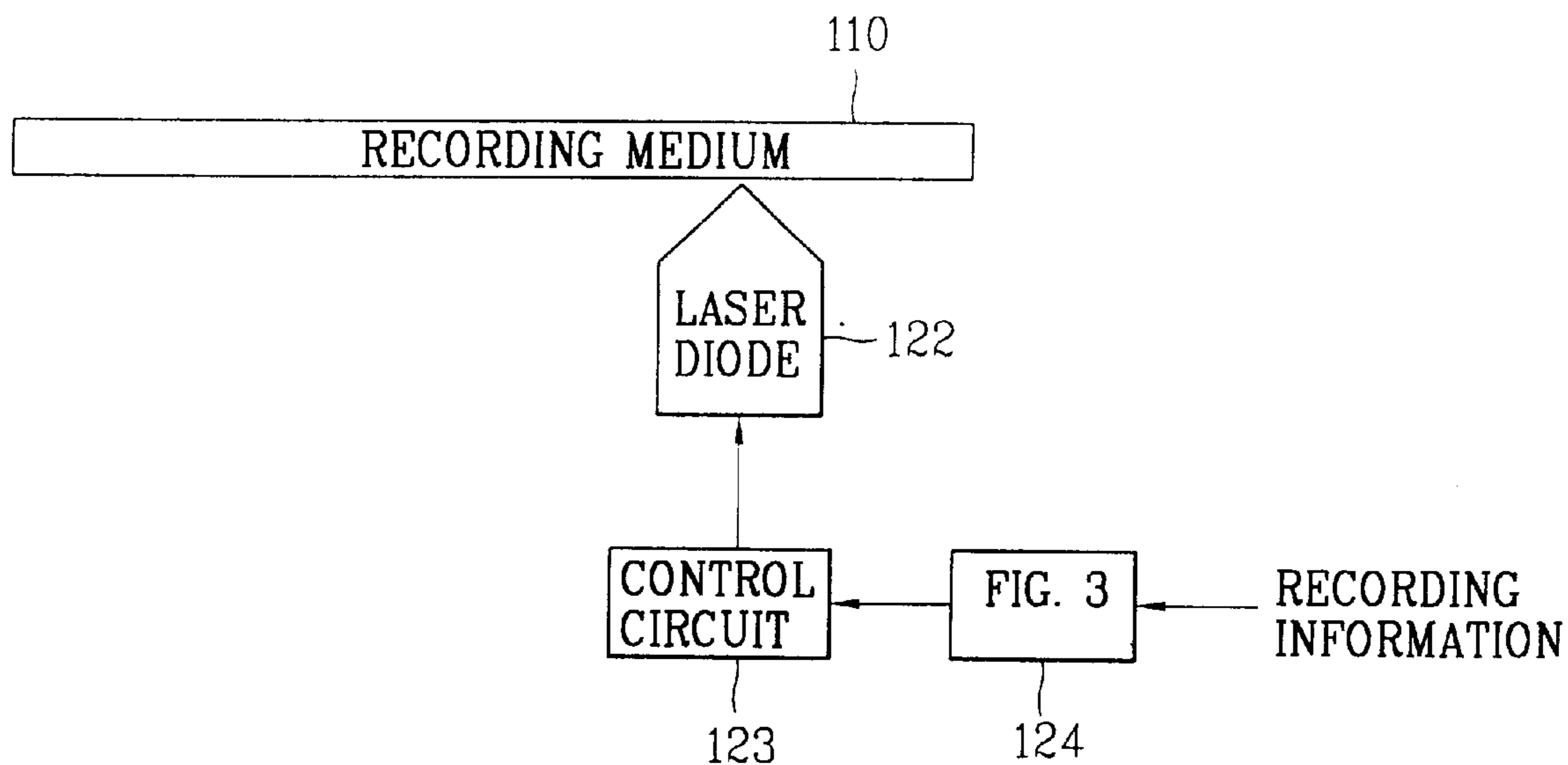


FIG. 7

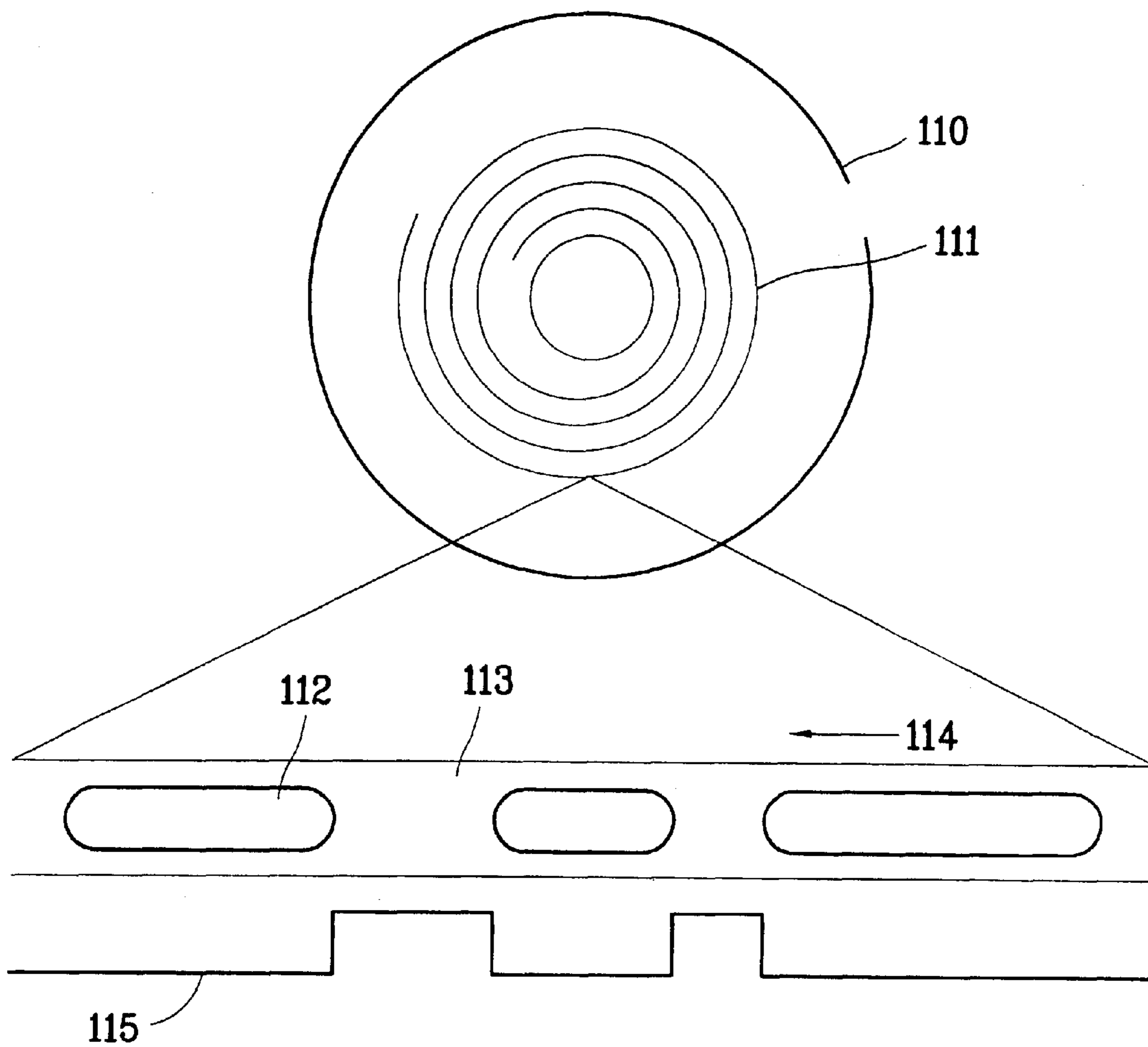


FIG. 8

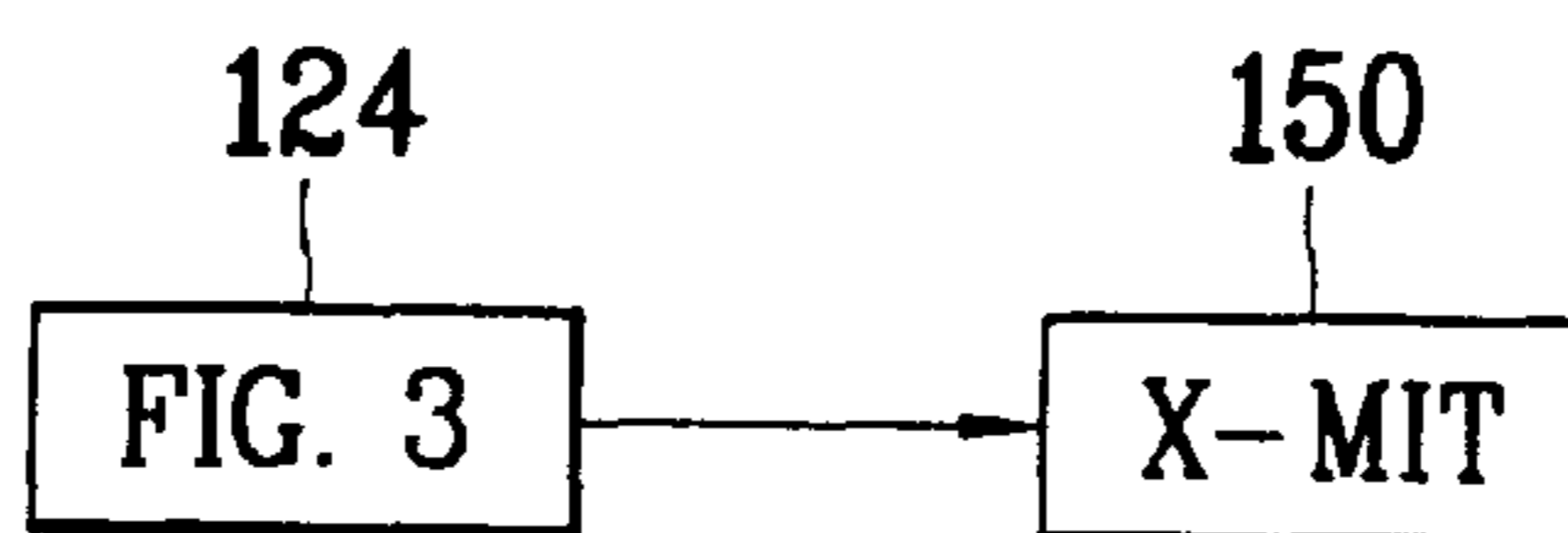


FIG. 9

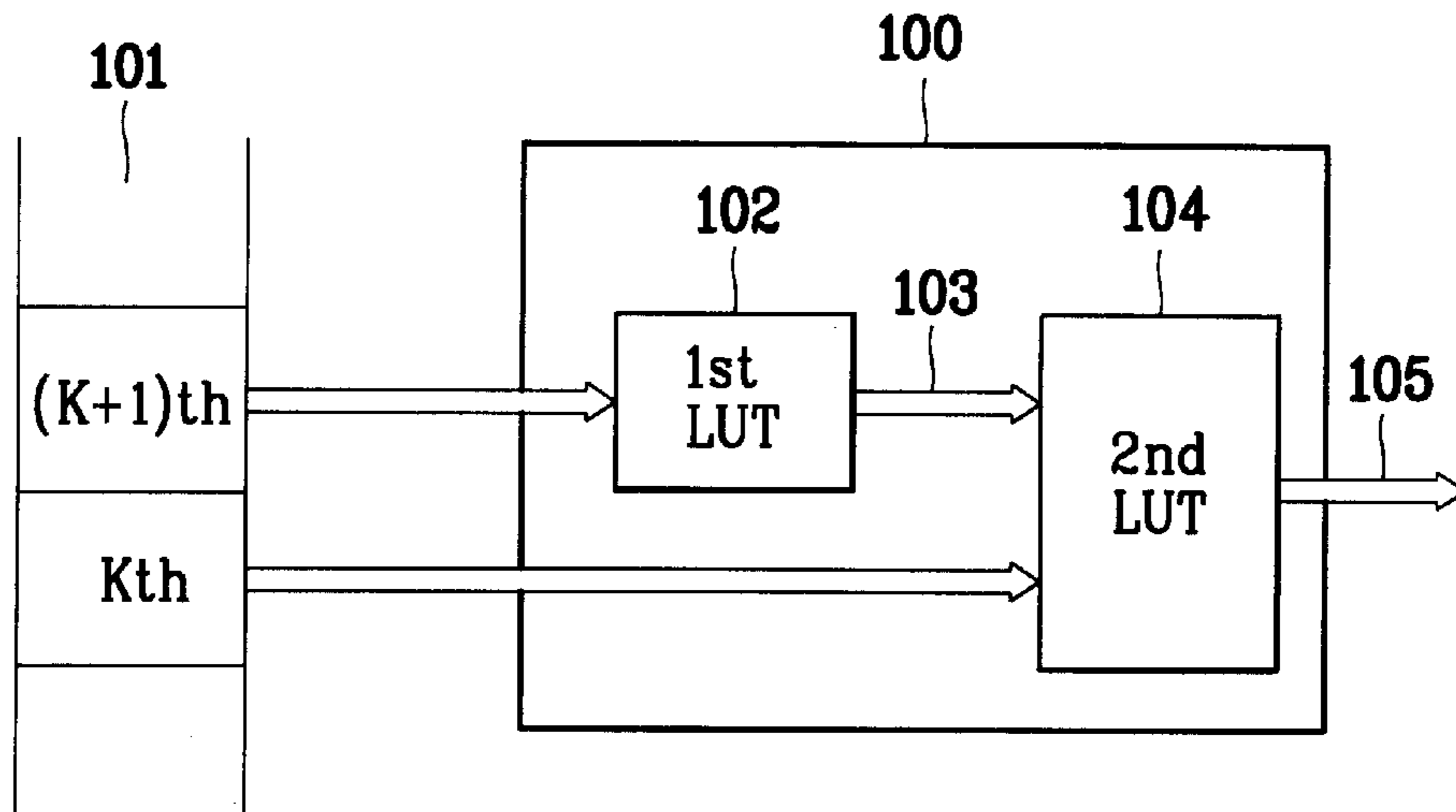


FIG. 10

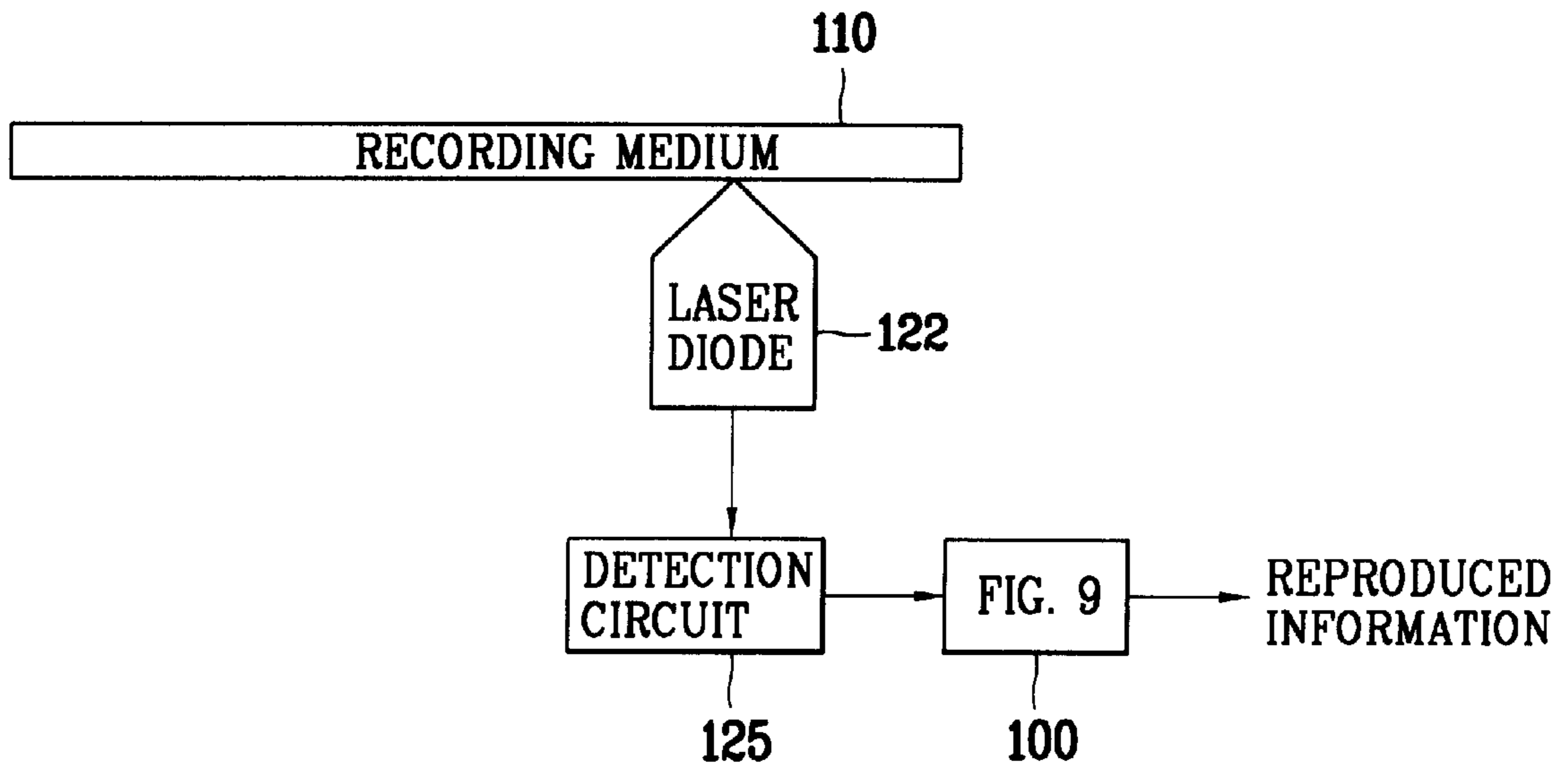




FIG. 11

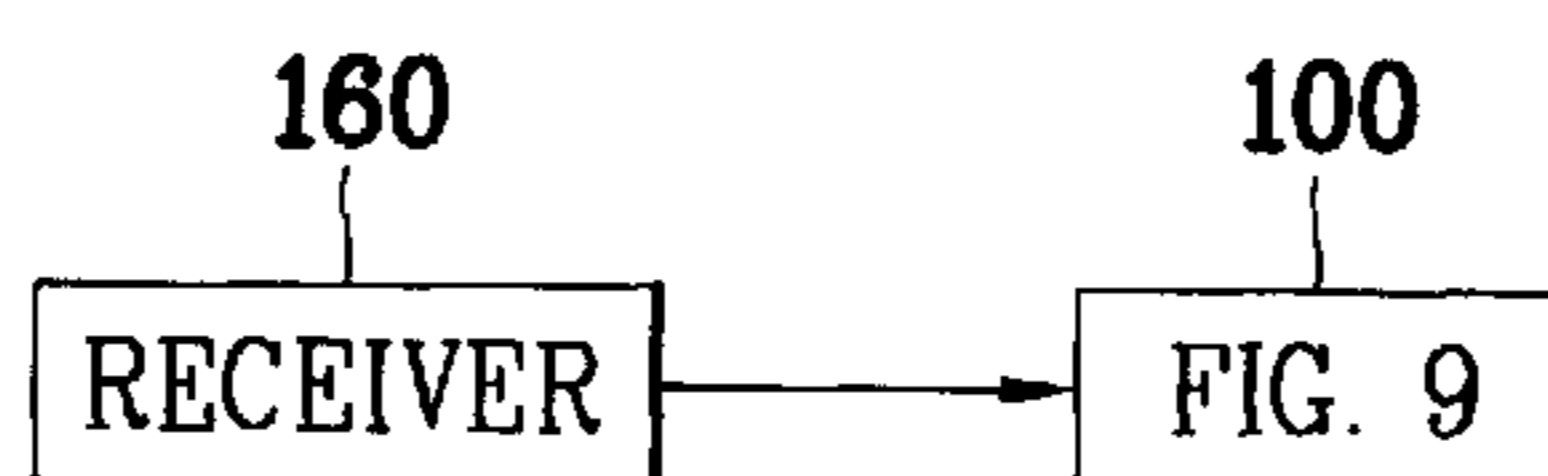


FIG. 12

SUBGROUP	1st KIND				2nd KIND			3rd KIND	
	STATE 1	STATE 2	STATE 3	STATE 4	STATE 5	STATE 6	STATE 7	STATE 8	STATE 9
E0000	152	152	152	119	0	0	0	0	0
E0010	65	65	65	5	0	0	60	0	0
E0001	70	70	70	160	0	0	10	0	0
E1000	0	0	0	0	152	152	102	0	0
E1010	0	0	0	0	65	65	59	0	0
E1001	0	0	0	0	70	70	137	0	0
E0100	0	0	0	0	0	0	0	152	125
E0110	0	0	0	0	0	0	0	65	53
E0101	0	0	0	0	0	0	0	71	119

FIG. 13A

Data	State 1		State 2		State 3		State 4		State 5	
	Channel bits	state	Channel bits	state	Channel bits	state	Channel bits	state	Channel bits	state
0	0000000000001000000	1	000001000100000000100	1	00010000010000100100	1	00100000100100001000	1	0010010010001001000	1
1	0000000000001000000	2	000001000100000000100	2	00010000010000100100	2	00100000100100001000	2	10000000000000000000	2
2	0000000000001000000	3	000001000100000000100	3	00010000010000100100	3	00100000100100001000	3	10000000000000000000	3
3	0000000000001000000	4	000001000100000000100	4	00010000010000100100	4	00100000100100001000	4	10000000000000000000	4
4	0000000000001000000	5	000001000100000000100	5	00010000010000100100	5	00100000100100001000	5	10000000000000000000	5
5	0000000000001000000	6	000001000100000000100	6	00010000010000100100	6	00100000100100001000	6	10000000000000000000	6
6	0000000000001000000	7	000001000100000000100	7	00010000010000100100	7	00100000100100001000	7	10000000000000000000	7
7	0000000000001000000	8	000001000100000000100	8	00010000010000100100	8	00100000100100001000	8	10000000000000000000	8
8	0000000000001000000	9	000001000100000000100	9	00010000010000100100	9	00100000100100001000	9	10000000000000000000	9
9	0000000000001000100	1	000001000100000000100	1	00010000010000100000	1	00100000100100001000	1	1000000000000000000	1
10	0000000000001000100	2	000001000100000000100	2	00010000010000100000	2	00100000100100001000	2	1000000000000000000	2
11	0000000000001000100	3	000001000100000000100	3	00010000010000100000	3	00100000100100001000	3	1000000000000000000	3
12	0000000000001000100	4	000001000100000000100	4	00010000010000100000	4	00100000100100001000	4	1000000000000000000	4
13	0000000000001000100	5	000001000100000000100	5	00010000010000100000	5	00100000100100001000	5	1000000000000000000	5
14	0000000000001000100	6	000001000100000000100	6	00010000010000100000	6	00100000100100001000	6	1000000000000000000	6
15	0000000000001000100	7	000001000100000000100	7	00010000010000100000	7	00100000100100001000	7	1000000000000000000	7
16	0000000000001000100	8	000001000100000000100	8	00010000010000100000	8	00100000100100001000	8	1000000000000000000	8
17	0000000000001000100	9	000001000100000000100	9	00010000010000100000	9	00100000100100001000	9	1000000000000000000	9
18	0000000000001001000	1	000001000100000000100	1	000100000100001000100	1	00100000100100001000	1	1000000000000000000	1
19	0000000000001001000	2	000001000100000000100	2	000100000100001000100	2	00100000100100001000	2	1000000000000000000	2
20	0000000000001001000	3	000001000100000000100	3	000100000100001000100	3	00100000100100001000	3	1000000000000000000	3
21	0000000000001001000	4	000001000100000000100	4	000100000100001000100	4	00100000100100001000	4	1000000000000000000	4
22	0000000000001001000	5	000001000100000000100	5	000100000100001000100	5	00100000100100001000	5	1000000000000000000	5
23	0000000000001001000	6	000001000100000000100	6	000100000100001000100	6	00100000100100001000	6	1000000000000000000	6
24	0000000000001001000	7	000001000100000000100	7	000100000100001000100	7	00100000100100001000	7	1000000000000000000	7
25	0000000000001001000	8	000001000100000000100	8	000100000100001000100	8	00100000100100001000	8	1000000000000000000	8
26	0000000000001001000	9	000001000100000000100	9	000100000100001000100	9	00100000100100001000	9	1000000000000000000	9
27	0000000000001000000	1	000001000100000000100	1	000100000100001000100	1	00100000100100001000	1	1000000000000000000	1
28	0000000000001000000	2	000001000100000000100	2	000100000100001000100	2	00100000100100001000	2	1000000000000000000	2
29	0000000000001000000	3	000001000100000000100	3	000100000100001000100	3	00100000100100001000	3	1000000000000000000	3
30	0000000000001000000	4	000001000100000000100	4	000100000100001000100	4	00100000100100001000	4	1000000000000000000	4
31	0000000000001000000	5	000001000100000000100	5	000100000100001000100	5	00100000100100001000	5	1000000000000000000	5

FIG. 13B

	State 6		State 7		State 8		State 9	
Data	Channel bits	state	Channel bits	state	Channel bits	state	Channel bits	state
0	10000100001000100000	1	10010000001001001000	1	00100100100100001001	1	01000100001000100000	1
1	10000100001000100000	2	10010000001001001000	2	01000000000000000000	2	01000100001000100000	2
2	10000100001000100000	3	10010000001001001000	3	01000000000000000000	3	01000100001000100000	3
3	10000100001000100000	4	10010000001001001000	4	01000000000000000000	4	01000100001000100000	4
4	10000100001000100000	5	10010000001001001000	5	01000000000000000000	5	01000100001000100000	5
5	10000100001000100000	6	10010000001001001000	6	01000000000000000000	6	01000100001000100000	6
6	10000100001000100000	7	10010000001001001000	7	01000000000000000000	7	01000100001000100000	7
7	10000100001000100000	8	10010000001001001000	8	01000000000000000000	8	01000100001000100000	8
8	10000100001000100000	9	10010000001001001000	9	01000000000000000000	9	01000100001000100000	9
9	10000100001000100100	1	10010000001000000000	1	01000000000000001000	1	01000100001000100100	1
10	10000100001000100100	2	10010000001000000000	2	01000000000000001000	2	01000100001000100100	2
11	10000100001000100100	3	10010000001000000000	3	01000000000000001000	3	01000100001000100100	3
12	10000100001000100100	4	10010000001000000000	4	01000000000000001000	4	01000100001000100100	4
13	10000100001000100100	5	10010000001000000000	5	01000000000000001000	5	01000100001000100100	5
14	10000100001000100100	6	10010000001000000000	6	01000000000000001000	6	01000100001000100100	6
15	10000100001000100100	7	10010000001000000000	7	01000000000000001000	7	01000100001000100100	7
16	10000100001000100100	8	10010000001000000000	8	01000000000000001000	8	01000100001000100100	8
17	10000100001000100100	9	10010000001000000000	9	01000000000000001000	9	01000100001000100100	9
18	10000100001001000000	1	1001000000100000001000	1	01000000000000001000	1	01000100001001000000	1
19	10000100001001000000	2	1001000000100000001000	2	01000000000000001000	2	01000100001001000000	2
20	10000100001001000000	3	1001000000100000001000	3	01000000000000001000	3	01000100001001000000	3
21	10000100001001000000	4	1001000000100000001000	4	01000000000000001000	4	01000100001001000000	4
22	10000100001001000000	5	1001000000100000001000	5	01000000000000001000	5	01000100001001000000	5
23	10000100001001000000	6	1001000000100000001000	6	01000000000000001000	6	01000100001001000000	6
24	10000100001001000000	7	1001000000100000001000	7	01000000000000001000	7	01000100001001000000	7
25	10000100001001000000	8	1001000000100000001000	8	01000000000000001000	8	01000100001001000000	8
26	10000100001001000000	9	1001000000100000001000	9	01000000000000001000	9	01000100001001000000	9
27	10000100001001000100	1	1001000000100000010000	1	0100000000000000100000	1	01000100001001000100	1
28	10000100001001000100	2	1001000000100000010000	2	0100000000000000100000	2	01000100001001000100	2
29	10000100001001000100	3	1001000000100000010000	3	0100000000000000100000	3	01000100001001000100	3
30	10000100001001000100	4	1001000000100000010000	4	0100000000000000100000	4	01000100001001000100	4
31	10000100001001000100	5	1001000000100000010000	5	0100000000000000100000	5	01000100001001000100	5



FIG. 15A

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
0	00000000100	1 00000100100	1 00010000000	2 00010000100	1 00100100010	1 10000100100	1 10010000000	2 01000000000	2 01000100100
1	00000000100	2 00000100100	2 00010000000	3 00010000100	2 00100100010	2 10000100100	2 10010000000	3 01000000000	3 01000100100
2	00000000100	3 00000100100	3 00010000000	4 00010000100	3 00100100010	3 10000100100	3 10010000000	4 01000000000	4 01000100100
3	00000000100	4 00000100100	4 00010000000	5 00010000100	4 00100100010	4 10000100100	4 10010000000	5 01000000000	5 01000100100
4	00000000100	5 00000100100	5 00010000000	6 00010000100	5 00100100010	5 10000100100	5 10010000000	6 01000000000	6 01000100100
5	00000000100	6 00000100100	6 00010000000	7 00010000100	6 00100100010	6 10000100100	6 10010000000	7 01000000000	7 01000100100
6	00000000100	7 00000100100	7 00010000000	8 00010000100	7 10000000000	7 10000100100	7 10010000000	8 01000000000	8 01000100100
7	00000000100	8 00000100100	8 00010000000	9 00010000100	8 10000000000	8 10000100100	8 10010000000	9 01000000000	9 01000100100
8	00000000100	9 00000100100	9 00100000000	2 00010000100	9 10000000000	9 10000100100	9 10010000000	1 01000000100	1 01000100100
9	00000001000	1 00001000000	1 00100000000	3 00100000100	1 10000000000	5 10000000000	1 10010000100	2 01000000100	2 01001000000
10	00000001000	2 00001000000	2 00100000000	4 00100000100	2 10000000000	6 10000000000	2 10010000100	3 01000000100	3 01001000000
11	00000001000	3 00001000000	3 00100000000	5 00100000100	3 10000000000	7 10000000000	3 10010000100	4 01000000100	4 01001000000
12	00000001000	4 00001000000	4 00100000000	6 00100000100	4 10000000000	8 10000000000	4 10010000100	5 01000000100	5 01001000000
13	00000001000	5 00001000000	5 00100000000	7 00100000100	5 10000000000	9 10000000000	5 10010000100	6 01000000100	6 01001000000
14	00000001000	6 00001000000	6 00100000000	8 00100000100	6 10000000000	1 10000000000	6 10010000100	7 01000000100	7 01001000000
15	00000001000	7 00001000000	7 00100000000	9 00100000100	7 10000000000	2 10000000000	7 10010000100	8 01000000100	8 01001000000
16	00000001000	8 00001000000	8 00100000000	1 00100000100	8 10000000000	3 10000000000	8 10010000100	9 01000000100	9 01001000000
17	00000001000	9 00001000000	9 00100000000	2 00100000100	9 10000000000	4 10000000000	9 10010000100	1 01000000100	1 01001000000
18	00000010000	1 00001000100	1 00010000100	3 00100001000	1 10000000100	5 10000000100	1 10010000100	2 01000001000	2 01001000100
19	00000010000	2 00001000100	2 00010000100	4 00100001000	2 10000000100	6 10000000100	2 10010000100	3 01000001000	3 01001000100
20	00000010000	3 00001000100	3 00010000100	5 00100001000	3 10000000100	7 10000000100	3 10010000100	4 01000001000	4 01001000100
21	00000010000	4 00001000100	4 00010000100	6 00100001000	4 10000000100	8 10000000100	4 10010000100	5 01000001000	5 01001000100
22	00000010000	5 00001000100	5 00010000100	7 00100001000	5 10000000100	9 10000000100	5 10010000100	6 01000001000	6 01001000100
23	00000010000	6 00001000100	6 00010000100	8 00100001000	6 10000000100	1 10000000100	6 10010000100	7 01000001000	7 01001000100
24	00000010000	7 00001000100	7 00010000100	9 00100001000	7 10000000100	2 10000000100	7 10010000100	8 01000001000	8 01001000100
25	00000010000	8 00001000100	8 00010000100	1 00100001000	8 10000000100	3 10000000100	8 10010000100	9 01000001000	9 01001000100
26	00000010000	9 00001000100	9 00010000100	2 00100001000	9 10000000100	4 10000000100	9 10010000100	1 01000001000	1 01001000100
27	00000010000	1 00001000100	1 00010000100	3 00100001000	1 10000000100	5 10000000100	1 10010000100	2 01000001000	2 01001000100
28	00000010000	2 00001000100	2 00010000100	4 00100001000	2 10000000100	6 10000000100	2 10010000100	3 01000001000	3 01001000100
29	00000010000	3 00001000100	3 00010000100	5 00100001000	3 10000000100	7 10000000100	3 10010000100	4 01000001000	4 01001000100
30	00000010000	4 00001000100	4 00010000100	6 00100001000	4 10000000100	8 10000000100	4 10010000100	5 01000001000	5 01001000100
31	00000010000	5 00001000100	5 00010000100	7 00100001000	5 10000000100	9 10000000100	5 10010000100	6 01000001000	6 01001000100

FIG. 15B

State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
32 0000010000	6 00001001000	6 00010010000	8 00100010000	6 10000010000	1 10001001000	6 10010010000	7 01000010000	7 01001001000
33 0000010000	7 00001001000	7 00010010000	9 00100010000	7 10000010000	2 10001001000	7 10010010000	8 01000010000	8 01001001000
34 0000010000	8 00001001000	8 00010000010	1 00100010000	8 10000010000	3 10001001000	8 10010010000	9 01000010000	9 01001001000
35 0000010000	9 00001001000	9 00010000010	2 00100010000	9 10000010000	4 10001001000	9 10010000010	1 01000100000	1 01001001000
36 00000000010	1 00000100010	1 00010000010	3 00100100000	1 10000010000	5 10000010010	1 10001000010	2 01000100000	2 01000100010
37 00000000010	2 00000100010	2 00010000010	4 00100100000	2 10000010000	6 10000010010	2 10001000010	3 01000100000	3 01000100010
38 00000000010	3 00000100010	3 00010000010	8 00100100000	3 10000010000	7 10000010010	3 10001000010	4 01000100000	4 01000100010
39 00000000010	4 00000100010	4 00010000010	9 00100100000	4 10000010000	8 10000010010	4 10001000010	8 01000100000	5 01000100010
40 00000000010	8 00000100010	8 00010010010	1 00100100000	5 10000010000	9 10000010010	8 10001000010	9 01000100000	6 01000100010
41 00000000010	9 00000100010	9 00010010010	2 00100100000	6 10000010000	1 10000010010	9 10001000010	1 01000100000	7 01000100010
42 00000010010	1 000001000010	1 00010010010	3 00100100000	7 10000010000	2 10000010010	1 10001000010	2 01000100000	8 01000100010
43 00000010010	2 000001000010	2 00010010010	4 00100100000	8 10000010000	3 10000010010	2 10001000010	3 01000100000	9 01000100010
44 00000010010	3 000001000010	3 00010010010	8 00100100000	9 10000010000	4 10000010010	3 10001000010	4 01000000010	1 01001000010
45 00000010010	4 000001000010	4 00010010010	9 00100100100	1 10000010000	5 10000010010	4 10001000010	8 01000000010	2 01001000010
46 00000010010	8 000001000010	8 00100000010	1 00100100100	2 10000010000	6 10000010010	8 10001000010	9 01000000010	3 01001000010
47 00000010010	9 000001000010	9 00100000010	2 00100100100	3 10000010000	7 10000010010	9 10001000010	1 01000000010	4 01001000010
48 00000001001	1 000001001001	1 00100000010	3 00100100100	4 10000010000	8 10000010010	1 10001000010	2 01000000010	8 01000010001
49 00000001001	2 000001001001	2 00100000010	4 00100100100	5 10000010000	9 10000010001	2 10001000010	3 01000000010	9 01000010001
50 00000001001	3 000001001001	3 00100000010	8 00100100100	6 10000000010	1 10000010001	3 10001000010	4 01000000010	1 01000010001
51 00000001001	4 000001001001	4 00100000010	9 00100100100	7 10000000010	2 10000010001	4 10001000010	8 01000000010	2 01000010001
52 00000001001	1 00010000001	1 00100000001	1 00100100100	8 10000000010	3 10000010001	1 10001000010	9 01000000010	3 01000010001
53 00000001001	2 00010000001	2 00100000001	2 00100100100	9 10000000010	4 10000010001	2 10001000001	1 01000000010	4 01000010001
54 00000001001	3 00010000001	3 00100000001	3 00100000010	1 10000000010	8 10000010001	3 10001000001	2 01000000010	8 01000010001
55 00000001001	4 00010000001	4 00100000001	4 00100000010	2 10000000010	9 10000010001	4 10001000001	3 01000000010	9 01000010001
56 00000001001	1 000100001001	1 001000001001	1 001000001001	3 10000000001	1 10001000001	1 10001000001	4 01000000001	1 01001000001
57 000000100001	2 000100001001	2 001000001001	2 001000001001	4 10000000001	2 10001000001	2 10001000001	1 01000000001	2 01001000001
58 000000100001	3 000100001001	3 001000001001	3 001000001001	8 10000000001	3 10001000001	3 10001000001	2 01000000001	3 01001000001
59 000000100001	4 000100001001	4 001000001001	4 001000001001	9 10000000001	4 10001000001	4 10001000001	3 01000000001	4 01001000001
60 000000100001	1 000100001001	1 001000001001	1 001000001001	1 10000000001	1 10001000001	1 10001000001	4 01000000001	4 01001000001
61 000000100001	2 000100001001	2 001000001001	2 001000001001	2 10000000001	2 10001000001	2 10001000001	1 01000000001	1 01001000001
62 000000100001	3 000100001001	3 001000001001	3 001000001001	3 10000000001	3 10001000001	3 10001000001	2 01000000001	2 01001000001
63 000000100001	4 000100001001	4 001000001001	4 001000001001	4 10000000001	4 10001000001	4 10001000001	3 01000000001	3 01001000001

FIG. 16

SUBGROUP	1st KIND				2nd KIND			3rd KIND	
	STATE 1	STATE 2	STATE 3	STATE 4	STATE 5	STATE 6	STATE 7	STATE 8	STATE 9
E0000	8	8	9	12	0	0	0	0	0
E0010	6	5	4	2	0	0	0	0	0
E0001	6	8	8	3	0	0	1	0	0
E1000	0	0	0	0	11	9	7	0	0
E1010	0	0	0	0	5	3	5	0	0
E1001	0	0	0	0	1	9	9	0	0
E0100	0	0	0	0	0	0	0	11	8
E0110	0	0	0	0	0	0	0	5	4
E0101	0	0	0	0	0	0	1	3	9

FIG. 17A

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
0	0000000100000	2 0000010010000	1 00010000000100	1 00100000000100	1 10000000000100	1 1000010010000	1 1001000001000	1 0100000000000	5 0100010001000
1	0000000100000	3 0000010010000	2 00010000000100	2 00100000000100	2 10000000000100	2 1000010010000	2 1001000001000	2 0100000000000	6 0100010001000
2	0000000100000	4 0000010010000	3 00010000000100	3 00100000000100	3 10000000000100	3 1000010010000	3 1001000001000	3 0100000000000	2 0100010001000
3	0000000100000	5 0000010010000	4 00010000000100	4 00100000000100	4 10000000000100	4 1000010010000	4 1001000001000	4 0100000000000	1 0100010001000
4	0000000100000	6 0000010010000	5 00010000000100	5 00100000000100	5 10000000000100	5 1000010010000	5 1001000001000	5 0100000000000	2 0100010001000
5	0000000100000	7 0000010010000	6 00010000000100	6 00100000000100	6 10000000000100	6 1000010010000	6 1001000001000	6 0100000000000	3 0100010001000
6	0000000100000	8 0000010010000	7 00010000000100	7 00100000000100	7 10000000000100	7 1000010010000	7 1001000001000	7 0100000000000	4 0100010001000
7	0000000100000	9 0000010010000	8 00010000000100	8 00100000000100	8 10000000000100	8 1000010010000	8 1001000001000	8 0100000000000	5 0100010001000
8	0000000100100	1 0000010010000	9 00010000000100	9 00100000000100	9 10000000000100	9 1000010010000	9 1001000001000	9 0100000000000	6 0100010001000
9	0000000100100	2 0000100000000	3 00010000000100	3 00100000000100	3 10000000000100	3 1000010010000	3 1001000001000	3 0100000000000	7 0100010001000
10	0000000100100	3 0000100000000	4 00010000000100	4 00100000000100	4 10000000000100	4 1000010010000	4 1001000001000	4 0100000000000	8 0100010001000
11	0000000100100	4 0000100000000	5 00010000000100	5 00100000000100	5 10000000000100	5 1000010010000	5 1001000001000	5 0100000000000	9 0100010001000
12	0000000100100	5 0000100000000	6 00010000000100	6 00100000000100	6 10000000000100	6 1000010010000	6 1001000001000	6 0100000000000	1 0100010001000
13	0000000100100	6 0000100000000	7 00010000000100	7 00100000000100	7 10000000000100	7 1000010010000	7 1001000001000	7 0100000000000	2 0100010001000
14	0000000100100	7 0000100000000	8 00010000000100	8 00100000000100	8 10000000000100	8 1000010010000	8 1001000001000	8 0100000000000	3 0100010001000
15	0000000100100	8 0000100000000	9 00010000000100	9 00100000000100	9 10000000000100	9 1000010010000	9 1001000001000	9 0100000000000	4 0100010001000
16	0000000100100	9 0000100000000	1 00010000000100	1 00010000000100	1 0000000001000	1 10001000000100	1 1001000001000	1 0100000000000	5 0100010001000
17	0000001000000	2 00001000000100	2 00010000000100	2 00010000000100	2 1000000001000	2 10001000000100	2 1001000001000	2 0100000000000	6 0100010001000
18	0000001000000	3 00001000000100	3 00010000000100	3 00010000000100	3 1000000001000	3 10001000000100	3 1001000001000	3 0100000000000	7 0100100000000
19	0000001000000	4 00001000000100	4 00010000000100	4 00010000000100	4 1000000001000	4 10001000000100	4 1001000001000	4 0100000000000	8 0100100000000
20	0000001000000	5 00001000000100	5 00010000000100	5 00010000000100	5 1000000001000	5 10001000000100	5 1001000001000	5 0100000000000	9 0100100000000
21	0000001000000	6 00001000000100	6 00010000000100	6 00010000000100	6 1000000001000	6 10001000000100	6 1001000001000	6 0100000000000	1 0100100000000
22	0000001000000	7 00001000000100	7 00010000000100	7 00010000000100	7 1000000001000	7 10001000000100	7 1001000001000	7 0100000000000	2 0100100000000
23	0000001000000	8 00001000000100	8 00010000000100	8 00010000000100	8 1000000001000	8 10001000000100	8 1001000001000	8 0100000000000	3 0100100000000
24	0000001000000	9 00001000000100	9 00010000000100	9 00010000000100	9 1000000001000	9 10001000000100	9 1001000001000	9 0100000000000	4 0100100000000
25	0000001000100	1 000010000001000	1 000100000001000	1 000100000001000	1 1000000001000	1 100010000001000	1 1001000001000	1 0100000000000	5 0100100000000
26	0000001000100	2 000010000001000	2 000100000001000	2 000100000001000	2 1000000001000	2 100010000001000	2 1001000001000	2 0100000000000	6 0100100000000
27	0000001000100	3 000010000001000	3 000100000001000	3 000100000001000	3 1000000001000	3 100010000001000	3 1001000001000	3 0100000000000	7 0100100000000
28	0000001000100	4 000010000001000	4 000100000001000	4 000100000001000	4 1000000001000	4 100010000001000	4 1001000001000	4 0100000000000	8 0100100000000
29	0000001000100	5 000010000001000	5 000100000001000	5 000100000001000	5 1000000001000	5 100010000001000	5 1001000001000	5 0100000000000	9 0100100000000
30	0000001000100	6 000010000001000	6 000100000001000	6 000100000001000	6 1000000001000	6 100010000001000	6 1001000001000	6 0100000000000	2 01001000000100
31	0000001000100	7 000010000001000	7 000100000001000	7 000100000001000	7 1000000001000	7 100010000001000	7 1001000001000	7 0100000000000	3 01001000000100



FIG. 17B

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
32	0000001000100	8 0000100001000	8 0001000100000	7 0010000100000	7 1000000100000	7 1000100001000	8 1001000100100	7 0100000100000	4 0100100000100
33	0000001000100	9 0000100001000	9 0001000100000	8 0010000100000	8 1000000100000	8 1000100001000	9 1001000100100	8 0100000100000	5 0100100000100
34	0000001000100	1 0000100001000	1 0001000100000	9 0010000100000	9 1000000100000	9 1000100001000	1 1001000100100	9 0100000100000	6 0100100000100
35	0000001000100	2 0000100001000	2 0001000100000	1 0010000100100	1 1000000100100	1 1000100001000	2 1001000100100	2 0100000100000	7 0100100000100
36	0000001000100	3 0000100001000	3 0001000100100	2 0010000100100	2 1000000100100	2 1000100001000	3 1001000100100	3 0100000100000	8 0100100000100
37	0000001000100	4 0000100001000	4 0001000100100	3 0010000100100	3 1000000100100	3 1000100001000	4 1001000100100	4 0100000100000	9 0100100000100
38	0000001000100	5 0000100001000	5 0001000100100	4 0010000100100	4 1000000100100	4 1000100001000	5 1001000100100	5 0100000100000	1 0100100000100
39	0000001000100	6 0000100001000	6 0001000100100	5 0010000100100	5 1000000100100	5 1000100001000	6 1001000100100	6 0100000100000	2 0100100000100
40	0000001000100	7 0000100001000	7 0001000100100	6 0010000100100	6 1000000100100	6 1000100001000	7 1001000100100	7 0100000100000	3 0100100000100
41	0000001000100	8 0000100001000	8 0001000100100	7 0010000100100	7 1000000100100	7 1000100001000	8 1001000100100	8 0100000100000	4 0100100000100
42	0000001000100	9 0000100001000	9 0001000100100	8 0010000100100	8 1000000100100	8 1000100001000	9 1001000100100	9 0100000100000	5 0100100000100
43	0000010000000	3 0000100100000	2 0001000100100	9 0010000100100	9 1000000100100	9 1000100001000	2 1001000100100	1 0100000100100	6 0100100001000
44	0000010000000	4 0000100100000	3 0001000100100	2 0010000100000	2 1000000100000	2 1000100001000	3 1001000100100	2 0100000100100	7 0100100001000
45	0000010000000	5 0000100100000	4 0001000100000	3 0010000100000	3 1000000100000	3 1000100001000	4 1001000100100	3 0100000100100	8 0100100001000
46	0000010000000	6 0000100100000	5 0001000100000	4 0010000100000	4 1000000100000	4 1000100001000	5 1001000100100	4 0100000100100	9 0100100001000
47	0000010000000	7 0000100100000	6 0001000100000	5 0010000100000	5 1000000100000	5 1000100001000	6 1001000100100	5 0100000100000	2 0100100001000
48	0000010000000	8 0000100100000	7 0001000100000	6 0010000100000	6 1000000100000	6 1000100001000	7 1001000100100	6 0100000100000	3 0100100001000
49	0000010000000	9 0000100100000	8 0001000100000	7 0010000100000	7 1000000100000	7 1000100001000	8 1001000100100	7 0100000100000	4 0100100001000
50	0000010000100	1 0000100100000	9 0001000100000	8 0010000100000	8 1000000100000	8 1000100001000	9 1001000100100	8 0100000100000	5 0100100001000
51	0000010000100	2 0000100100000	1 0001000100000	9 0010000100000	9 1000000100000	9 1000100001000	1 1001000100100	9 0100000100000	6 0100100001000
52	0000010000100	3 0000100100000	2 0001000100000	1 0010000100000	1 1000000100000	1 1000100001000	2 1001000100100	1 0100000100000	7 0100100001000
53	0000010000100	4 0000100100000	3 0001000100000	2 0010000100000	2 1000000100000	2 1000100001000	3 1001000100100	2 0100000100000	8 0100100001000
54	0000010000100	5 0000100100000	4 0001000100000	3 0010000100000	3 1000000100000	3 1000100001000	4 1001000100100	3 0100000100000	9 0100100001000
55	0000010000100	6 0000100100000	5 0001000100000	4 0010000100000	4 1000000100000	4 1000100001000	5 1001000100100	4 0100000100000	1 0100100001000
56	0000010000100	7 0000100100000	6 0001000100000	5 0010000100000	5 1000000100000	5 1000100001000	6 1001000100100	5 0100000100000	2 0100100001000
57	0000010000100	8 0000100100000	7 0001000100000	6 0010000100000	6 1000000100000	6 1000100001000	7 1001000100100	6 0100000100000	3 0100100001000
58	0000010000100	9 0000100100000	8 0001000100000	7 0010000100000	7 1000000100000	7 1000100001000	8 1001000100100	7 0100000100000	4 0100100001000
59	0000010001000	1 0000100100100	9 0001000100010	8 0010000100010	8 1000000100010	8 1000100001000	9 1001000100100	8 0100000100010	5 0100100001000
60	0000010001000	2 0000100000000	4 0001000000000	9 0010000000000	9 1000000000000	9 1000100000000	1 1001000000000	9 0100000100010	6 0100100001000
61	0000010001000	3 0001000000000	5 0001000000000	1 0010000000000	1 1000000000000	1 1000100000000	2 1001000000000	1 0100000100010	7 0100100001000
62	0000010001000	4 0001000000000	6 0001000000000	2 0010000000000	2 1000000000000	2 1000100000000	3 1001000000000	2 0100000100010	8 0100100001000
63	0000010001000	5 0001000000000	7 0001000000000	3 0010000000000	3 1000000000000	3 1000100000000	4 1001000000000	3 0100000100010	9 0100100001000

FIG. 17C

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
64	000010001000	6 000100000000	8 000100100100	4 001000100100	4 100000100100	4 100100000000	8 100010010001	4 010000100100	1 010010010010
65	000010001000	7 000100000000	9 000100100100	5 001000100100	5 100000100100	5 100100000000	9 100010010001	8 010000100100	2 010010010010
66	000010001000	8 000010010001	1 000100100100	6 001000100100	6 100000100100	6 100100000000	1 100010010001	9 010000100100	3 010010010010
67	000010001000	9 000010010001	2 000100100100	7 001000100100	7 100000100100	7 100100000000	2 100100000000	1 010000100100	4 010010010010
68	000000100010	1 000010010001	3 000100100100	8 001000100100	8 100000100100	8 100100000000	3 100100000000	2 010000100100	5 010010010010
69	000000100010	2 000010010001	4 000100100100	9 001000100100	9 100000100100	9 100100000000	4 100100000000	3 010000100100	6 010010010010
70	000000100010	3 000010010001	8 001000000000	5 001001000000	3 100001000000	3 100100000000	5 100100000000	4 010000100100	7 010001001001
71	000000010001	4 000010010001	9 001000000000	6 001001000000	4 100010000000	4 100100000000	6 100100000000	8 010000100100	8 010001001001
72	000000010001	8 000100000000	1 001000000000	7 001001000000	5 100001000000	5 100100000000	7 100100000000	9 010000100100	9 010001001001
73	000000010001	9 000100000000	2 001000000000	8 001001000000	6 100001000000	6 100100000000	8 100100000000	1 010001000000	3 010001000000
74	000000010001	1 000100000000	3 001000000000	9 001001000000	7 100001000000	7 100100000000	9 100100000000	2 010001000000	4 010001000000
75	000000010001	2 000100000000	4 001000000000	1 001001000000	8 100001000000	8 100001000000	1 100100000000	3 010001000000	5 010010000000
76	000000010001	3 000100000000	8 001000000000	2 001001000000	9 100001000000	9 100001000000	2 100100000000	4 010001000000	6 010010000000
77	000000010001	4 000100000000	9 001000000000	3 001001000000	1 100001000000	1 100001000000	3 100100000000	8 010001000000	7 010010000000
78	000000010001	8 000100000000	1 001000000000	4 001001000000	2 100001000000	2 100001000000	4 100100000000	9 010001000000	8 010010000000
79	000000010001	9 000100000000	2 001000000000	8 001001000000	3 100001000000	3 100001000000	8 100100000000	1 010001000000	9 010010000000
80	000001000001	1 000100000001	3 001000000001	9 001001000001	4 100001000001	4 100001000001	9 100100000001	2 010001000001	1 010010000001
81	000001000001	2 000100000001	4 001000000001	1 001001000001	5 100001000001	5 100001000001	1 100100000001	3 010001000001	2 010010000001
82	000001000001	3 000100000001	8 001000000001	2 001001000001	6 100001000001	6 100001000001	2 100100000001	4 010001000001	3 010010000001
83	000001000001	4 000100000001	9 001000000001	3 001001000001	7 100001000001	7 100001000001	3 100100000001	8 010001000001	4 010010000001
84	000001000001	8 000100000001	1 001000000001	4 001001000001	8 100001000001	8 100001000001	4 100100000001	9 010001000001	5 010010000001
85	000001000001	9 000100000001	2 001000000001	8 001001000001	9 100001000001	9 100001000001	8 100100000001	1 010001000001	6 010010000001
86	000001000001	1 000100000001	3 001000000001	9 001001000001	1 100001000001	1 100001000001	9 100100000001	2 010001000001	7 010010000001
87	000001000001	2 000100000001	4 001000000001	1 001001000001	2 100001000001	2 100001000001	1 100100000001	3 010001000001	8 010010000001
88	000001000001	3 000100000001	8 001000000001	2 001001000001	3 100001000001	3 100001000001	2 100100000001	4 010001000001	9 010010000001
89	000001000001	4 000100000001	9 001000000001	3 001001000001	4 100001000001	4 100001000001	3 100100000001	8 010000000001	1 010010000001
90	000001000001	8 000100000001	1 001000000001	4 001001000001	5 100001000001	5 100001000001	4 100100000001	9 010000000001	2 010010000001
91	000001000001	9 000100000001	2 001000000001	8 001001000001	6 100001000001	6 100001000001	8 100100000001	1 010000000001	3 010010000001
92	000010000001	1 000100000001	3 001000000001	9 001001000001	7 100001000001	7 100001000001	9 100100000001	2 010000000001	4 010010000001
93	000010000001	2 000100000001	4 001000000001	1 001001000001	8 100001000001	8 100001000001	1 100100000001	3 010000000001	5 010001000001
94	000010000001	3 000100000001	8 001000000001	2 001001000001	9 100001000001	9 100001000001	2 100100000001	4 010000000001	6 010001000001
95	000010000001	4 000100000001	9 001000000001	3 001001000001	1 100000000001	1 100000000001	3 100010000001	1 010000000001	7 010001000001

FIG. 17D

	State 1	State 2	State 3	State 4	State 5	State 6	State 7	State 8	State 9
Data	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state	Channel bits state
96	0000100000010	8 0000100000000	1 0010001000010	4 0010010010000	2 1000000000010	2 1000000001001	4 1000100010001	2 010000010010	2 0100001000001
97	0000100000010	9 0000100000001	2 0010001000010	8 0010010010000	3 1000000000010	3 1000000001001	1 1000100010001	3 0100000010010	3 0100001001001
98	0000100010010	1 0000100000001	3 0010001000010	9 0010010010000	4 1000000000010	4 1000000001001	2 1000100010001	4 0100000010010	4 0100001001001
99	0000100010010	2 0000100000001	4 0001001000001	1 0010010010000	5 1000000000010	8 1000000001001	3 1000100100001	1 0100000010010	8 0100001001001
100	0000100010010	3 0000100001001	1 0001001000001	2 0010010010000	6 1000000000010	9 1000000001001	4 1000100100001	2 0100000010010	9 0100001001001
101	0000100010010	4 0000100001001	2 0001001000001	3 0010010010000	7 1000000001001	1 1000000010001	1 1000100100001	3 0100000010010	1 0100001000001
102	0000100010010	8 0000100001001	3 0001001000001	4 0010010010000	8 1000000001001	2 1000000010001	2 1000100100001	4 0100000010010	2 0100001000001
103	0000100010010	9 0000100001001	4 0001001000001	1 0010010010000	9 1000000001001	3 1000000010001	3 1001000000001	1 0100000010010	3 0100001000001
104	0000000100001	1 0000100010001	1 0001001001001	2 0010010000010	1 1000000010001	4 1000000010001	4 1001000000001	2 0100000010010	4 0100001000001
105	0000000100001	2 0000100010001	2 0001001001001	3 0010010000010	2 1000000010001	1 1000000010001	1 1001000000001	3 0100000010010	8 0100001000101
106	0000000100001	3 0000100010001	3 0001001001001	4 0010010000010	3 1000000010001	2 1000000010001	2 1001000000001	4 0100000010010	9 0100001000101
107	0000000100001	4 0000100010001	4 0001000000001	1 0010010000010	4 1000000010001	3 1000000010001	3 1001000000101	1 0100000010010	1 0100001000101
108	0000001000001	1 0000100100001	1 0001000000001	2 0010010000010	8 1000000010001	2 1000000010001	4 1001000000101	2 0100000010010	2 0100001000101
109	0000001000001	2 0000100100001	2 0001000000001	3 0010010000010	9 1000000010001	3 1000000010001	1 1001000000101	3 0100000010010	3 0100001000101
110	0000001000001	3 0000100100001	3 0001000000001	4 0010010000010	1 1000000010001	4 1000000010001	2 1001000000101	4 0100000010010	4 0100001000101
111	0000001000001	4 0000100100001	4 0001000000001	1 0010010000010	2 1000000010001	8 1000000010001	3 1001000000101	1 0100000010010	8 0100001000101
112	0000001001001	1 0001000000001	1 0001000000001	2 0010010000010	3 1000000010001	9 1000000010001	4 1001000000101	2 0100000010010	9 0100001000101
113	0000001001001	2 0001000000001	2 0001000000001	3 0010010000010	4 1000000010001	1 1000000010001	1 1001000000101	3 0100000010010	1 0100100000001
114	0000001001001	3 0001000000001	3 0001000000001	4 0010010000010	8 1000000010001	2 1000000010001	2 1001000000101	4 0100000010010	2 0100100000001
115	0000001001001	4 0001000000001	4 0001000000001	1 0010010000010	9 1000000010001	3 1000000010001	3 1001000000101	1 0100000010010	3 0100100000001
116	0000001000001	1 0001000000101	1 0010000001001	2 0010010000010	1 1000000010001	4 1000000010001	4 1001000000101	2 0100000010010	4 0100100000001
117	0000001000001	2 0001000001001	2 0010000001001	3 0010010000010	2 1000000010001	8 1000000010001	1 1001000000101	3 0100000010010	8 0100100001001
118	0000001000001	3 0001000001001	3 0010000001001	4 0010010000010	3 1000000010001	9 1000000010001	2 1001000000101	4 0100000010010	9 0100100001001
119	0000001000001	4 0001000001001	4 0010000001001	1 0010010000010	4 1000000010001	1 1000000010001	3 1001000000101	1 0100000010010	1 0100100001001
120	0000001000101	1 0001000001001	1 0010000001001	2 0010010000010	2 1000000010001	3 1000000010001	4 1001000000101	2 0100000010010	2 0100100001001
121	0000001000101	2 0001000001001	2 0010000001001	3 0010010000010	3 1000000010001	4 1000000010001	1 1001000000101	3 0100000010010	3 0100100001001
122	0000001000101	3 0001000001001	3 0010000001001	4 0010010000010	4 1000000010001	1 1000000010001	2 1001000000101	4 0100000010010	4 0100100001001
123	0000001000101	4 0001000001001	4 0010000001001	1 0010010000010	8 1000000010001	3 1000000010001	3 1001000000101	1 0100000010010	1 0100100001001
124	0000001000001	1 0001000001001	1 0010000001001	2 0010010000010	9 1000000010001	4 1000000010001	4 1001000000101	2 0100000010010	2 0100100001001
125	0000001000001	2 0001000001001	2 0010000001001	3 0010010000010	1 1000000010001	9 1000000010001	1 1001000000101	3 0100000010010	3 0100100001001
126	0000001000001	3 0001000001001	3 0010000001001	4 0010010000010	2 1000000010001	1 1000000010001	2 1001000000101	4 0100000010010	4 0100100001001
127	0000001001001	4 0001000010001	4 0010000001001	1 0010010000010	3 1000000010001	4 1000000010001	3 1001000000001	1 0100000010001	1 0100100001001

**METHOD AND APPARATUS FOR CODING INFORMATION, METHOD AND APPARATUS FOR DECODING CODED INFORMATION, METHOD OF FABRICATING A RECORDING MEDIUM, THE RECORDING MEDIUM AND MODULATED SIGNAL**

**BACKGROUND OF THE INVENTION**

1. Field of the Invention

The present invention relates to coding information, and more particularly, to a method and apparatus for coding information having improved information density. The present invention further relates to producing a modulated signal from the coded information, producing a recording medium from the coded information, and the recording medium itself. The present invention still further relates to a method and apparatus for decoding coded information, and decoding coded information from a modulated signal and/or a recording medium.

2. Description of the Background Art

When data is transmitted through a transmission line or recorded onto a recording medium such as a magnetic disc, an optical disc or a magneto-optical disc, the data is modulated into code matching the transmission line or the recording medium prior to the transmission or recording.

Run length limited codes, generically designated as  $(d, k)$  codes, have been widely and successfully applied in modern magnetic and optical recording systems. Such codes, and means for implementing such codes are described by K. A. Schouhamer Immink in the book entitled "Codes for Mass Data Storage Systems" (ISBN 90-74249-23-X, 1999). Run length limited codes are extensions of earlier non return to zero recording codes, where binary recorded "zeros" are represented by no (magnetic flux) change in the recording medium, while binary "ones" are represented by transitions from one direction of recorded flux to the opposite direction.

In a  $(d, k)$  code, the above recording rules are maintained with the additional constraints that at least  $d$  "zeros" are recorded between consecutive "ones", and no more than  $k$  "zeros" are recorded between consecutive "ones". The first constraint arises to obviate intersymbol interference occurring because of pulse crowding of the reproduced transitions when a series of "ones" are contiguously recorded. The second constraint arises to ensure recovering a clock from the reproduced data by "locking" a phase locked loop to the reproduced transitions. If there is too long an unbroken string of contiguous "zeros" with no interspersed "ones", the clock regenerating phase-locked-loop will fall out of synchronism. In, for example, a  $(2,7)$  code there is at least two "zeros" between recorded "ones", and there are no more than seven recorded contiguous "zeros" between recorded "ones".

The series of encoded bits is converted, via a modulo-2 integration operation, to a corresponding modulated signal formed by bit cells having a high or low signal value. A "one" bit is represented in the modulated signal by a change from a high to a low signal value or vice versa, and a "zero" bit is represented by the lack of change in the modulated signal.

The information conveying efficiency of such codes is typically expressed as a rate, which is the quotient of the number of bits  $(m)$  in the information word to the number of bits  $(n)$  in the code word (i.e.,  $m/n$ ). The theoretical maximum rate of a code, given values of  $d$  and  $k$ , is called the

Shannon capacity. FIG. 1 tabulates the Shannon capacity  $C(d,k)$  for  $d=2$  versus  $k$ . As shown, for a  $(2,7)$  code, the Shannon capacity,  $C(2,7)$ , has a value of 0.5174. This means that a  $(2,7)$  code cannot have a rate larger than 0.5174. The practical implementation of codes requires that the rate be a rational fraction, and to date the above  $(2,7)$  code has a rate  $\frac{1}{2}$ . This rate of  $\frac{1}{2}$  is slightly less than the Shannon capacity of 0.5174, and the code is therefore a highly efficient one. To achieve the  $\frac{1}{2}$  rate, 1 unconstrained data bit is mapped into 2 constrained encoded bits.

$(2,7)$  codes having a rate of  $\frac{1}{2}$  and means for implementing associated encoders and decoders are known in the art. U.S. Pat. No. 4,115,768 entitled "Sequential Encoding and Decoding of Variable Word Length, Fixed Rate Data Codes", issued in the names of Eggenberger and Hodges, discloses an encoder whose output sequences satisfy the imposed runlength constraints.

However, a demand exists for even more efficient codes so that, for example, the information density on a recording medium or over a transmission line can be increased.

**SUMMARY OF THE INVENTION**

In the converting method and apparatus according to the present invention,  $m$ -bit information words are converted into  $n$ -bit code words at a rate greater than  $\frac{1}{2}$ . Consequently, the same amount of information can be recorded in less space, and information density increased.

In the present invention,  $n$ -bit code words are divided into a first type, a second type and a third type, and into coding states of a first kind, a second kind and a third kind such that an  $m$ -bit information word is converted into an  $n$ -bit code word of the first, second or third kind if the previous  $m$ -bit information word was converted into an  $n$ -bit code word of the first type and is converted into an  $n$ -bit code word of the first or third kind, if the previous  $m$ -bit information word was converted into an  $n$ -bit code word of the second type and is converted into an  $n$ -bit code word of the first kind, if the previous  $m$ -bit information word was converted into an  $n$ -bit code word of the third type. Further, sets of code words belonging to the different coding states do not contain any code words in common. In one embodiment,  $n$ -bit code words of the first type end in "00",  $n$ -bit code words of the second type end in "10",  $n$ -bit code words of the third type end in "01",  $n$ -bit code words belonging to the states of the first kind start with "00",  $n$ -bit code words belonging to the states of the second kind start with "00", "01" or "10", and  $n$ -bit code words belonging to the states of the third kind start with "00" or "01". Furthermore, in the embodiments according to the present invention, the  $n$ -bit code words satisfy a  $dk$ -constraint to  $(2,k)$  such that a minimum of 2 zeros and a maximum of  $k$  zeros falls between consecutive ones.

In other embodiments of the present invention, the coding device and method according to the present invention are employed to record information on a recording medium and create a recording medium according to the present invention.

In still other embodiments of the present invention, the coding device and method according to the present invention are further employed to transmit information.

In the decoding method and apparatus according to the present invention,  $n$ -bit code words created according to the coding method and apparatus are decoded into  $m$ -bit information words. The decoding involves determining the state of a next  $n$ -bit code word, and based on the state determination, the current  $n$ -bit code word is converted into an  $m$ -bit information word.

In other embodiments of the present invention, the decoding device and method according to the present invention are employed to reproduce information from a recording medium.

In still other embodiments of the present invention, the decoding device and method according to the present invention are employed to receive information transmitted over a medium.

### BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will become more fully understood from the detailed description given herein below and the accompanying drawings which are given by way of illustration only, wherein like reference numerals designate corresponding parts in the various drawings, and wherein:

FIG. 1 tabulates the Shannon capacity  $C(d,k)$  for  $d=2$  versus  $k$ ;

FIG. 2 shows an example of how the code words in the various subgroups are allocated into the various states in the first embodiment;

FIG. 3 shows an embodiment for a coding device according to the invention;

FIGS. 4A–4B show a complete translation table according to the first embodiment for converting 6-bit information words into 11-bit code words;

FIG. 5 illustrates the conversion of a series of information words into a series of code words using the translation table of FIGS. 4A–4B;

FIG. 6 illustrates an embodiment of a recording device according to the present invention;

FIG. 7 illustrates a recording medium and modulated signal according to the present invention;

FIG. 8 illustrates a transmission device according to the present invention;

FIG. 9 illustrates a decoding device according to the present invention;

FIG. 10 illustrates a reproducing device according to the present invention;

FIG. 11 illustrates a receiving device according to the present invention;

FIG. 12 shows an example of how the code words in the various subgroups are allocated in to the various states in the second embodiment;

FIGS. 13A–13B show the beginning portion of a translation table according to the second embodiment for converting 11-bit information words into 20-bit code words;

FIG. 14 shows an example of how the code words in the various subgroups are allocated into the various states in the third embodiment;

FIGS. 15A–15B show a translation table according to the third embodiment for converting 6-bit information words into 11-bit code words;

FIG. 16 shows an example of how the code words in the various subgroups are allocated into the various states in the fourth embodiment;

FIGS. 17A–17D show a translation table according to the fourth embodiment for converting 7-bit information words into 13-bit code words.

### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The general coding method according to the present invention will be described followed by a specific first

embodiment of the coding method. Next, the general decoding method according to the present invention will be described in the context of the first embodiment. The various apparatuses according to the present invention will then be described. Specifically, the coding device, recording device, transmission device, decoding device, reproducing device and receiving device according to the present invention will be described. Afterwards, additional coding embodiments according to the present invention will be described.

### CODING METHOD

According to the present invention, an  $m$ -bit information word is converted into an  $n$ -bit code word such that the rate of  $m/n$  is greater than  $1/2$ . The code words are divided into first, second and third types wherein the first type includes code words ending with “00”, the second type includes code words ending with “10” and the third type includes code words ending with “01”. As a result, the code words of the first type are divided into three subgroups **E0000**, **E1000** and **E0100**, code words of the second type are divided into three subgroups **E0010**, **E1010** and **E0110**, and the code words of the third type are divided into three subgroups **E0001**, **E1001** and **E0101**. Code word subgroup **E0000** includes code words that start with “00” and end with “00”, code word subgroup **E1000** includes code words that start with “10” and end with “00”, code word subgroup **E0100** includes code words that start with “01” and end with “00”. Code word subgroup **E0010** includes code words that start with “00” and end with “10”, code word subgroup **E1010** includes code words that start with “10” and end with “10”, code word subgroup **E0110** includes code words that start with “01” and end with “10”. Code word subgroup **E0001** includes code words that start with “00” and end with “01”, code word subgroup **E1001** includes code words that start with “10” and end with “01”, code word subgroup **E0101** includes code words that start with “01” and end with “01”.

The code words are also divided into at least one state of a first kind, at least one state of a second kind, and at least one state of a third kind. States of the first kind include code words that only start with “00” and states of the second kind include code words that start with one of the “00”, “01” and “10”, and states of the third kind include code words that start with either “00” or “01”.

Further, sets of code words belonging to the different coding states do not contain any code words in common. In other words, different states can not include the same code word.

### CODING METHOD ACCORDING TO A FIRST EMBODIMENT

In a first preferred embodiment of the present invention, 6-bit information words are converted into 11-bit code words. The code words satisfy a  $(2,k)$  constraint, an  $d$  are divided into  $4(r1)$  states of the first kind,  $3(r2)$  states of the second kind, and  $2(r3)$  states of the third kind (total of  $r=r1+r2+r3=9$  states). In order to reduce the  $k$ -constraint, a code word, namely, “000000000000” is barred from the encoding tables.

To perform encoding, each 11-bit code word in each state is associated with a coding state direction. The state direction indicates the next state from which to select a code word in the encoding process. The state directions are assigned to code words such that code words that end with a “00” (i.e. code words in subgroups **E0000**, **E1000** and **E0100**) have associated state directions that indicate any of the  $r=9$  states, while code words that end with a “10” (i.e., code words in

subgroups **E0010**, **E1010** and **E0110**) have associated state directions that only indicate one of the states of the first kind or the third kind. Furthermore, code words that end with a “01” (i.e., code words in subgroups **E0001**, **E1001** and **E0101**) have associated state directions that only indicate one of the states of the first kind. This ensures that the  $d=2$  constraint will be satisfied.

Furthermore, while, as explained in more detail below, the same code word can be assigned to different information words in the same state, different states cannot include the same code word. In particular code words in subgroups **E0000**, **E1000** and **E0100** can be assigned 9 times to different information words within one state, while code words in subgroups **E0010**, **E1010** and **E0110** can be assigned 6 times to different information words within one state. Furthermore, code words in subgroups **E0001**, **E1001** and **E0101** can be assigned 4 times to different information words within one state. As there are 18 code words in subgroup **E0000**, 13 code words in subgroup **E1000** and 9 code words in subgroup **E0100**, there are 360 ( $9 \times (18+13+9)$ ) “code word—state direction” combinations for code words of the first type. There are 9 code words in subgroup **E0010**, 6 code words in subgroup **E1010** and 4 code words in subgroup **E0110**, so that there are 114 ( $6 \times (9+6+4)$ ) “code word—state direction” combinations for code words of the second type. There are 11 code words in subgroup **E0001**, 9 code words in subgroup **E1001** and 6 code words in subgroup **E0101**, so that there are 104 ( $4 \times (11+9+6)$ ) “code word—state direction” combinations for code words of the third type. In total  $360+114+104=578$  “code word—state direction” combinations exist.

For  $m$ -bit information words, there are a total of  $2^m$  possible information words. So, for 6-bit information words,  $2^6=64$  information words exist. Because there are nine states in this encoding embodiment, 9 times  $64=576$  of the “code word—state direction” combinations are needed. This leaves  $578-576=2$  remaining combinations.

The available code words in the various subgroups are distributed over the states of the first, the second and third kind in compliance with the restrictions discussed above. FIG. 2 shows an example of how the code words in the various subgroups are allocated in this embodiment to the various states. As shown in FIG. 2, in this example, states **1**, **2**, **3** and **4** are states of the first kind, and states **5**, **6** and **7** are states of the second kind, and states **8** and **9** are states of the third kind. Taking the subgroup **E0000** of size 18 as an example, subgroup **E0000** has 6 code words in states **1** and **4** code word in each of states **2**, **3** and **4**. And, taking state **1** as an example, in state **1** the number of “code word—state direction” combinations is  $9 \times 6 + 6 \times 1 + 4 \times 1 = 64$ , which means that 6-bit information words can be assigned. Remember, each code word of the first type can be assigned any one of the nine different states as a state directions, and therefore used nine times within a state; while each code word of the second type can only be assigned one of the six states of the first kind and the third kind as a state direction because of the  $d=2$  restriction, and therefore used six times within a state. Furthermore, each code word of the third type can only be assigned one of the four states of the first kind as a state direction because of the  $d=2$  restriction, and therefore used four times within a state.

It can be verified that from any of the  $r=9$  coding states shown in FIG. 2 there at least 64 information words that can be assigned to code words, which is enough to accommodate 6-bit information words. In the manner described above any random series of 6-bit information words can be uniquely converted to a series of code words.

FIGS. 4A–4B show a complete translation table according to this embodiment for converting 6-bit information words into 11-bit code words. Included in the translation table of FIGS. 4A–4B are the state direction assigned to each code word. Specifically, in FIGS. 4A–4B, the first column shows the decimal notation of the information words. The second, fourth, sixth, eighth, tenth, twelfth, fourteenth, sixteenth, and eighteenth columns show the code words (also referred to in the art as channel bits) assigned to the information words in states **1**, **2**, **3**, **4**, **5**, **6**, **7**, **8** and **9**, respectively. The third, fifth, seventh, ninth, eleventh, thirteenth, fifteenth, seventeenth and nineteenth columns show by way of the respective digits **1**, **2**, **3**, **4**, **5**, **6**, **7**, **8** and **9**, the state direction of the associated code words in the second, fourth, sixth, eighth, tenth, twelfth, fourteenth, sixteenth, and eighteenth columns, respectively.

The conversion of a series of information words into a series of code words will be further explained with reference to FIG. 5. The first column of FIG. 5 shows from top to bottom a series of successive 6-bit information words, and the second column shows in parenthesis the decimal values of these information words. The third column “state” is the coding state that is to be used for the conversion of the information word. The “state” is laid down when the preceding code word was delivered (i.e., the state direction of the preceding code word). The fourth column “code words” includes the code words assigned to the information words according to the translation table of FIGS. 4A–B. The fifth column “next state” is the state direction associated with the code word in the fourth column and is also determined according to the translation table of FIGS. 4A–B.

The first word from the series of information words shown in the first column of FIG. 5 has a word value of “1” in decimal notation. Let us assume that the coding state is state **1** (**S1**) when the conversion of the series of information words is initiated. Therefore the first word is translated into code word “0000000100” according to the state **1** set of code words from the translation table. At the same time the next state becomes state **2** (**S2**) because the state direction assigned to code word “0000000100” representing decimal value **1** in state **1** is state **2**. This means that the next information word (decimal value “3”) is going to be translated using the code words in state **2**. Consequently, the next information word, having a decimal value of “3”, is translated into code word “00001000100”. Similar to the manner described above, the information words having the decimal values “5”, “12” and “19” are converted.

#### Decoding Method

Hereinafter, decoding of  $n$ -bit code words (in this example 11-bit words) received from a recording medium will be further explained with reference to FIGS. 4A–4B. For the purposes of description, assume that the word values of a series of successive code words received from, for example, a recording medium are “0000000100”, “0001001000”, “10000100100”. From the translation table of FIGS. 4A–4B, it is found that the first code word “0000000100” is assigned to the information words “9”, “10”, “11”, “12”, “13”, “14”, “15”, “16”, and “17” and state directions **1**, **2**, **3**, **4**, **5**, **6**, **7**, **8** and **9**, respectively. The next code word value is “0001001000”, and belongs to the set of code words in state **3**. This means that the first code word “0000000100” had a state direction of **3**. The first code word “0000000100” with a state direction of **3** represents the information word having a decimal value of “11”. Therefore, it is determined that the first code word represents information word “0000000100” having a decimal value of “11”.

Furthermore, the third code word "10000100100" is a member of state 6. Therefore, it is determined in the same manner as above that the second code word "00010010000" represents the information word having the decimal value "14". In the same manner other code words can be decoded. It is noted that both the current code word and the next code words are observed to decode the current code word into a unique information word.

#### Coding Device

FIG. 3 shows an embodiment for a coding device 124 according to the invention. The coding device 124 converts m-bit information words into n-bit code words, where the number of different coding states r is represented by s bits. For example, when the number of coding states  $r=9$ , s equals 4. As shown, the coding device 124 includes a converter 50 for converting (m+s) binary input signals to (n+s) binary output signals. In a preferred embodiment, the converter 50 includes a read only memory (ROM) storing a translation table according to at least one embodiment of the present invention and address circuitry for addressing the translation table based on the m+s binary input signals. However, instead of a ROM, the converter 50 can include a combinatorial logic circuit producing the same results as the translation table according to at least one embodiment of the present invention.

From the inputs of the converter 50, m inputs are connected to a first bus 51 for receiving m-bit information words. From the outputs of the converter 50, n outputs are connected to a second bus 52 for delivering n-bit code words. Furthermore, s inputs are connected to an s-bit third bus 53 for receiving a state word that indicates the instantaneous coding state. The state word is delivered by a buffer memory 54 including, for example, s flip-flops. The buffer memory 54 has s inputs connected to a fourth bus 55 for receiving a state direction to be loaded into the buffer memory 54 as the state word. For delivering the state directions to be loaded in the buffer memory 54, the s outputs of the converter 50 are used.

The second bus 52 is connected to the parallel inputs of a parallel-to-serial converter 56, which converts the code words received over the second bus 52 to a serial bit string. A signal line 57 supplies the serial bit string to a modulator circuit 58, which converts the bit string into a modulated signal. The modulated signal is then delivered over a line 60. The modulator circuit 58 is any well-known circuit for converting binary data into a modulated signal such as a modula-2 integrator.

For the purposes of synchronizing the operation of the coding device, the coding device includes a clock generating circuit (not shown) of a customary type for generating clock signals for controlling timing of, for example, the parallel/serial converter 58 and the loading of the buffer memory 54.

In operation, the converter 50 receives m-bit information words and an s-bit state word from the first bus 51 and the third bus 53, respectively. The s-bit state word indicates the state in the translation table to use in converting the m-bit information word. Accordingly, based on the value of the m-bit information word, the n-bit code word is determined from the code words in the state identified by the s-bit state word. Also, the state direction associated with the n-bit code word is determined. The state direction, namely, the value thereof is converted into an s-bit binary word; or alternatively, the state directions are stored in the translation table as s-bit binary words. The converter 50 outputs the n-bit code word on the second bus 52, and outputs the s-bit

state direction on fourth bus 55. The buffer memory 54 stores the s-bit state direction as a state word, and supplies the s-bit state word to the converter 50 over the third bus 53 in synchronization with the receipt of the next m-bit information word by the converter 50. This synchronization is produced based on the clock signals discussed above in any well-known manner.

The n-bit code words on the second bus 52 are converted to serial data by the parallel/serial converter 56, and then the serial data is converted into a modulated signal by the modulator 58.

The modulated signal may then undergo further processing for recordation or transmission.

#### Recording Device

FIG. 6 shows a recording device for recording information that includes the coding device 124 according to the present invention as shown in FIG. 3. As shown in FIG. 6, m-bit information is converted into a modulated signal through the coding device 124. The modulated signal produced by the coding device 124 is delivered to a control circuit 123. The control circuit 123 may be any conventional control circuit for controlling an optical pick-up or laser diode 122 in response to the modulated signal applied to the control circuit 123 so that a pattern of marks corresponding to the modulated signal are recorded on the recording medium 110.

FIG. 7 shows by way of example, a recording medium 110 according to the invention. The recording medium 110 shown is a read-only memory (ROM) type optical disc. However, the recording medium 110 of the present invention is not limited to a ROM type optical disk, but could be any type of optical disk such as a write-once read-many (WORM) optical disk, random accessible memory (RAM) optical disk, etc. Further, the recording medium 110 is not limited to being an optical disk, but could be any type of recording medium such as a magnetic disk, a magneto-optical disk, a memory card, magnetic tape, etc.

As shown in FIG. 7, the recording medium 110 according to one embodiment of the present invention includes information patterns arranged in tracks 111. Specifically, FIG. 7 shows an enlarged view of a track 111 along a direction 114 of the track 111. As shown, the track 111 includes pit regions 112 and non-pit regions 113. Generally, the pit and non-pit regions 112 and 113 represent constant signal regions of the modulated signal 115 (zeros in the code words) and the transitions between pit and non-pit regions represent logic state transitions in the modulated signal 115 (ones in the code words).

As discussed above, the recording medium 110 may be obtained by first generating the modulated signal and then recording the modulated signal on the recording medium 110. Alternatively, if the recording medium is an optical disc, the recording medium 110 can also be obtained with well-known mastering and replica techniques.

#### Transmission Device

FIG. 8 shows a transmission device for transmitting information that includes the coding device 124 according to the present invention as shown in FIG. 3. As shown in FIG. 8, m-bit information words are converted into a modulated signal through the coding device 124. A transmitter 150 then further processes the modulated signal, to convert the modulated signal into a form for transmission depending on the communication system to which the transmitter belongs, and

transmits the converted modulated signal over a transmission medium such as air (or space), optical fiber, cable, a conductor, etc.

#### Decoding Device

FIG. 9 illustrates a decoder according to the present invention. The decoder performs the reverse process of the converter of FIG. 3 and converts n-bit code words of the present invention into m-bit information words. As shown, the decoder 100 includes a first look-up table (LUT) 102 and a second LUT 104. The first and second LUTs 102 and 104 store the translation table used to create the n-bit code words being decoded. Where K refers to time, the first LUT 102 receives the (K+1)th n-bit code word and the second LUT 104 receives the output of the first LUT 102 and the Kth n-bit code word. Accordingly, the decoder 100 operates as a sliding block decoder. At every block time instant the decoder 100 decodes one n-bit code word into one m-bit information word and proceeds with the next n-bit code word in the serial data (also referred to as the channel bit stream).

In operation, the first LUT 102 determines the state of the (K+1)th code word from the stored translation table, and outputs the state to the second LUT 104. So the output of the first LUT 102 is a binary number in the range of 1, 2, . . . , r (where r denotes the number of states in the translation table). The second LUT 104 determines the possible m-bit information words associated with Kth code word from the Kth code word using the stored translation table, and then determines the specific one of the possible m-bit information words being represented by the n-bit code word using the state information from the first LUT 102 and the stored translation table.

For the purposes of further explanation only, assume the n-bit code words are 11-bit code words produced using the translation table of FIGS. 4A–4B. Then, referring to FIG. 5, if the (K+1)th 11-bit code word is “00001000100” the first LUT 102 determines the state as state 2. Furthermore, if the Kth 11-bit code word is “00000000100”, then the second LUT 104 determines that the Kth 11-bit code word represents one of the 6-bit information words having a decimal value of 0, 1, 2, 3, 4, 5, 6, 7, 8 or 9. And, because the next state or state direction of state 2 is supplied by the first LUT 102, the second LUT 104 determines that the Kth 11-bit code word represents the 6-bit information word having a decimal value of 1 because the 11-bit code word “00000000100” associated with a state direction of 2 represents the 6-bit information word having a decimal value of 1.

#### Reproducing Device

FIG. 10 illustrates a reproducing device that includes the decoder 100 according to the present invention as shown in FIG. 9. As shown, the reading device includes an optical pick-up 122 of a conventional type for reading a recording medium 110 according to the invention. The recording medium 110 may be any type of recording medium such as discussed previously. The optical pickup 122 produces an analog read signal modulated according to the information pattern on the recording medium 110. A detection circuit 125 converts this read signal in conventional fashion into a binary signal of the form acceptable to the decoder 100. The decoder 100 decodes the binary signal to obtain the m-bit information words.

#### Receiving Device

FIG. 11 illustrates a receiving device that includes the decoder 100 according to the present invention as shown in

FIG. 9. As shown, the receiving device includes a receiver 160 for receiving a signal transmitted over a medium such as air (or space), optical fiber, cable, a conductor, etc. The receiver 160 converts the received signal into a binary signal of the form acceptable to the decoder 100. The decoder 100 decodes the binary signal to obtain the m-bit information words.

#### Coding Method According to a Second Embodiment

FIGS. 12 and 13A–13B illustrate another embodiment of the present invention. According to this embodiment, the greater than  $\frac{1}{2}$  rate is achieved by converting 11-bit information words into 20-bit code words; wherein the number of coding states r equals 9, and 4 of the coding states are coding states of the first kind and 3 of the coding states are coding states of the second kind and 2 of the coding states are coding state of the third kind. Also, the code words satisfy a (2,k) constraint. FIG. 12 corresponds to FIG. 2 of the first embodiment, and illustrates the division of code words among the states in this second embodiment.

As described above, code words that end with a “00”, i.e. code words in subgroups E0000, E1000, and E0100, are allowed to enter any of the r=9 states, while code words that end with a “10” i.e. code words in subgroups E0010, E1010 and E0110, may only enter the states of the first kind or third kind (State 1 to State 4 or State 8 to State 9). Furthermore, code words that end with a “01” i.e. code words in subgroups E0001, E1001 and E0101, may only enter the states of the first kind (State 1 to State 4).

Therefore, code words in subgroups E0000, E1000 and E0100 can be assigned 9 times to different information words, while code words in subgroups E0010, E1010 and E0110 can be assigned 6 times to different information words, and code words in subgroups E0001, E1001 and E0101 can be assigned 4 times to different information words. Referring to FIG. 12, subgroup E0000 has 152 code words in state 1, and the subgroup E0010 has 65 code words in state 1, and the subgroups E0001 has 70 code words in state 1. So the number of “code words—state direction” combinations is  $(9 \times 152) + (6 \times 65) + (4 \times 75) = 2,058$ , which means that 11-bit information words ( $2^{11} = 2,048$ ) can be assigned. It can be verified that from any of the r=9 coding states there at least 2,048 information words that can be assigned to code words, which is enough to accommodate 11-bit information words.

FIGS. 13A–13B illustrate the beginning portion of the translation table for this second embodiment in the same fashion that FIGS. 4A–4B illustrated the translation table for the first embodiment.

#### Coding Method According to a Third Embodiment

FIGS. 14 and 15A–15B illustrate another embodiment of the present invention. According to this embodiment, the greater than  $\frac{1}{2}$  rate is achieved by converting 6-bit information words into 11-bit code words; wherein the number of coding states r equals 9, and 4 of the coding states are coding states of the first kind and 3 of the coding states are coding states of the second kind and 2 of the coding states are coding state of the third kind, similar to those of the first embodiment. Also, the code words satisfy a (2,k) constraint. FIG. 14 corresponds to FIG. 2 of the first embodiment, and illustrates the division of code words among the states in this third embodiment. It can be verified that from any of the r=9 coding states there at least 64 information words that can be assigned to code words, which is enough to accommodate 6-bit information words.



## 11

FIGS. 15A–15B illustrate a translation table for this third embodiment in the same fashion that FIGS. 4A–4B illustrated the translation table for the first embodiment.

## Coding Method According to a Fourth Embodiment

FIGS. 16 and 17A–17D illustrate another embodiment of the present invention. According to this embodiment, the greater than  $\frac{1}{2}$  rate is achieved by converting 7-bit information words into 13-bit code words; wherein the number of coding states  $r$  equals 9, and 4 of the coding states are coding states of the first kind and 3 of the coding states are coding states of the second kind and 2 of the coding states are coding state of the third kind, similar to those of the first embodiment. Also, the code words satisfy a  $(2,k)$  constraint. FIG. 16 corresponds to FIG. 2 of the first embodiment, and illustrates the division of code words among the states in this fourth embodiment. It can be verified that from any of the  $r=9$  coding states there at least 128 information words that can be assigned to code words, which is enough to accommodate 7-bit information words.

FIGS. 17A–17D illustrate a translation table for this fourth embodiment in the same fashion that FIGS. 4A–4B illustrated the translation table for the first embodiment.

The invention has been described in detail with particular reference to preferred embodiments thereof, but it will be understood that variations and modifications can be effected within the spirit and scope of the invention.

What is claimed is:

1. A method of converting, comprising: receiving  $m$ -bit information words, where  $m$  is an integer; converting the  $m$ -bit information words into  $n$ -bit code words, where  $n$  is an integer greater than  $m$ , the  $n$ -bit code words being divided into a first type, a second type and a third type, and into coding states of a first kind, a second kind and a third kind such that an  $m$ -bit information word is converted into an  $n$ -bit code word of the first, second or third kind if the previous  $m$ -bit information word was converted into an  $n$ -bit code word of the first type and is converted into an  $n$ -bit code word of the first or third kind, if the previous  $m$ -bit information word was converted into an  $n$ -bit code word of the second type and is converted into an  $n$ -bit code word of the first kind, if the previous  $m$ -bit information word was converted into an  $n$ -bit code word of the third type, wherein sets of code words belonging to the different coding states do not contain any code words in common.
2. The method of claim 1, wherein the converting step converts the  $m$ -bit information words into  $n$ -bit code words that satisfy a  $dk$ -constraint, where  $d$  indicates a minimum number of zeros between consecutive ones in the  $n$ -bit code words and  $k$  indicates a maximum number of zeros between consecutive ones in the  $n$ -bit code words.
3. The method of claim 2, wherein  $m/n$  is greater than  $\frac{1}{2}$ , and  $d=2$ .
4. The method of claim 2, wherein  $d=2$ .
5. The method of claim 2, wherein the  $n$ -bit code words are divided into  $r_1$  coding states of the first kind,  $r_2$  coding states of the second kind and  $r_3$  coding states of the third kind, where  $r_1$ ,  $r_2$ , and  $r_3$  are integers greater than or equal to 1, and each of the  $r_1$ ,  $r_2$  and  $r_3$  coding states have  $n$ -bit code words different from the  $n$ -bit code words in the other  $r_1$ ,  $r_2$  and  $r_3$  coding states.
6. The method of claim 5, wherein  $m/n$  is greater than  $\frac{1}{2}$ ,  $d=2$ ,  $r_1=4$ ,  $r_2=3$  and  $r_3=2$ .
7. The method of claim 5, wherein  $r_1=4$ ,  $r_2=3$  and  $r_3=2$ .

## 12

8. The method of claim 5, wherein  $r_1+r_2+r_3$  equals 9.

9. The method of claim 1, wherein the  $n$ -bit code words of the first type end in “00”, the  $n$ -bit code words of the second type end in “10” and the  $n$ -bit code words of the third type end in “01”.

10. The method of claim 9, wherein the  $n$ -bit code words in a coding state of the first kind start with “00”, the  $n$ -bit code words in a coding state of the second kind start with “00”, “01” or “10”, and the  $n$ -bit code words in a coding state of the third kind start with “00” or “01”.

11. The method of claim 1, wherein the converting step converts at a coding rate of  $m/n$ , which is greater than  $\frac{1}{2}$ .

12. The method of claim 1, wherein  $m$  is equal to 6 and  $n$  is equal to 11.

13. The method of claim 1, further comprising:

generating a modulated signal from the  $n$ -bit code words.

14. The method of claim 13, further comprising:

recording the modulated signal in a recording medium.

15. The method of claim 13, further comprising:

transmitting the modulated signal.

16. The method of claim 1, wherein the converting step converts the  $m$ -bit information words into the  $n$ -bit code words using a translation table.

17. The method of claim 1, wherein  $m$  is equal to 11 and  $n$  is equal to 20.

18. The method of claim 1, wherein  $m$  is equal to 7 and  $n$  is equal to 13.

19. A method of converting, comprising:

receiving  $m$ -bit information words, where  $m$  is an integer;

converting the  $m$ -bit information words into  $n$ -bit code

words that satisfy a  $dk$ -constraint, where  $n$  is an integer

greater than  $m$ ,  $d$  indicates a minimum number of zeros

between consecutive ones in the  $n$ -bit code words and

$k$  indicates a maximum number of zeros between

consecutive ones in the  $n$ -bit code words, the  $n$ -bit code

words being divided into a first type, a second type and

a third type and into coding states of a first kind, a

second kind and a third kind an  $m$ -bit information word

is converted into an  $n$ -bit code word of the first, second

or third kind if the previous  $m$ -bit information word

was converted into an  $n$ -bit code word of the first type

and is converted into an  $n$ -bit code word of the first or

third kind, if the previous  $m$ -bit information word was

converted into an  $n$ -bit code word of the second type

and is converted into an  $n$ -bit code word of the first

kind, if the previous  $m$ -bit information word was con-

verted into an  $n$ -bit code word of the third type, wherein

the  $n$ -bit code words of the first type end in “00”, the

$n$ -bit code words of the second type end in “10” and the

$n$ -bit code words of the third type end in “01”, and  $n$ -bit

code words in a coding state of the first kind start with

“00”, the  $n$ -bit code words in a coding state of the

second kind start with “00” “01”, or “10”, and the  $n$ -bit

code words in a coding state of the third kind start with

“00” or “01”, wherein sets of code words belonging to

the different coding states do not contain any code

words in common.

20. A coding device, comprising:

a converter receiving  $m$ -bit information words, where  $m$

is an integer and converting the  $m$ -bit information

words into  $n$ -bit code words, where  $n$  is an integer

greater than  $m$ , the  $n$ -bit code words being divided into

a first type, a second type and a third type, and into

coding states of a first kind, a second kind and a third

kind such that an  $m$ -bit information word is converted

into an  $n$ -bit code word of the first, second or third kind

if the previous m-bit information word was converted into an n-bit code word of the first type and is converted into an n-bit code word of the first or third kind, if the previous m-bit information word was converted into an n-bit code word of the second type and is converted into an n-bit code word of the first kind, if the previous m-bit information word was converted into an n-bit code word of the third type, wherein sets of code words belonging to the different coding states do not contain any code words in common.

**21.** The coding device of claim **20**, wherein the converter receives a coding state with each m-bit information word and converts the m-bit information word into the n-bit code word based on the coding state.

**22.** The coding device of claim **21**, further comprising:  
a buffer supplying the coding state to the converter; and wherein the converter determines the coding state for the next m-bit information word as part of the converting process, and stores the determined coding state in the buffer.

**23.** The coding device of claim **22**, wherein the converter converts the m-bit information word into the n-bit code word and determines the coding state using a translation table.

**24.** The coding device of claim **20**, further comprising:  
a modulator generating a modulated signal from the n-bit code words.

**25.** The coding device of claim **24**, further comprising:  
a recording device recording the modulated signal in a recording medium.

**26.** The coding device of claim **24**, further comprising:  
a transmitter transmitting the modulated signal.

**27.** A method of manufacturing a recording medium, comprising:

converting m-bit information words into n-bit code words, where n is an integer greater than m, the n-bit code words being divided into a first type, a second type and a third type and into coding states of a first kind, a second kind and a third kind such that an m-bit information word is converted into an n-bit code word of the first, second or third kind if the previous m-bit information word was converted into an n-bit code word of the first type and is converted into an n-bit code word of the first or third kind, if the previous m-bit information word was converted into an n-bit code word of the second type and is converted into an n-bit code word of the first kind, if the previous m-bit information word was converted into an n-bit code word of the third type, wherein sets of code words belonging to the different coding states do not contain any code words in common;

generating a modulated signal from the n-bit code words; and

recording the modulated signal in a recording medium.

**28.** A recording medium having a modulated signal recorded in a track, the modulated signal including signal portions representing n-bit code words, where n is an integer, each n-bit code word representing an m-bit information word, where m is an integer less than n, the n-bit code words being divided into a first type, a second type and a third type and into coding states of a first kind, a second kind and a third kind such that an m-bit information word is converted into an n-bit code word of the first, second or third kind if the previous m-bit information word was converted into an n-bit code word of the first type and is converted into an n-bit code word of the first or third kind, if the previous

m-bit information word was converted into an n-bit code word of the second type and is converted into an n-bit code word of the first kind, if the previous m-bit information word was converted into an n-bit code word of the third type, wherein sets of code words belonging to the different coding states do not contain any code words in common.

**29.** The recording medium of claim **28**, wherein the signal portions represent the n-bit code words such that each successive n-bit code word partially instructs a reproducing device on which of at least two m-bit information words are represented by each previous n-bit code word.

**30.** A modulated signal, comprising:

signal portions representing n-bit code words, where n is an integer, each n-bit code word representing an m-bit information word, where m is an integer less than n, the n-bit code words being divided into a first type, a second type and a third type and into coding states of a first kind, a second kind and a third kind such that an m-bit information word is converted into an n-bit code word of the first, second or third kind if the previous m-bit information word was converted into an n-bit code word of the first type and is converted into an n-bit code word of the first or third kind, if the previous m-bit information word was converted into an n-bit code word of the second type and is converted into an n-bit code word of the first kind, if the previous m-bit information word was converted into an n-bit code word of the third type, wherein sets of code words belonging to the different coding states do not contain any code words in common.

**31.** The modulated signal of claim **30**, wherein the signal portions represent the n-bit code words such that each successive n-bit code word partially instructs a reproducing device on which of at least two m-bit information words are represented by each previous n-bit code word.

**32.** A method of decoding, comprising:

receiving n-bit code words, where n is an integer;

converting the n-bit code words into m-bit information words, where m is an integer less than n, the n-bit code words being divided into a first type, a second type and a third type and into coding states of a first kind, a second kind and a third kind such that an m-bit information word is converted into an n-bit code word of the first, second or third kind if the previous m-bit information word was converted into an n-bit code word of the first type and is converted into an n-bit code word of the first or third kind, if the previous m-bit information word was converted into an n-bit code word of the second type and is converted into an n-bit code word of the first kind, if the previous m-bit information word was converted into an n-bit code word of the third type, wherein sets of code words belonging to the different coding states do not contain any code words in common.

**33.** The method of claim **32**, wherein the n-bit code words are divided into **r1** coding states of the first kind, **r2** coding states of the second kind and **r3** coding states of the third kind, where **r1**, **r2** and **r3** are integers greater than or equal to 1, and each of the **r1**, **r2** and **r3** coding states have n-bit code words different from the n-bit code words in the other **r1**, **r2** and **r3** coding states.

**34.** The method of claim **33**, wherein the converting step determines to which of the p and q coding states a next n-bit code word belongs, and converts a current n-bit code word into an m-bit information word based on the determined coding state.

**35.** The method of claim **34**, wherein at least one of the **r1**, **r2** and **r3** coding states includes more than one of a same

n-bit code word, the same n-bit code word maps to more than one of the m-bit information words, and each same n-bit code word has a different state direction associated therewith, each state direction indicating a next one of the r1, r2 and r3 coding states from which to obtain the next n-bit code word when converting the m-bit information words into the n-bit code words.

36. The method of claim 35, wherein the n-bit code words satisfy a dk-constraint, where d indicates a minimum number of zeros between consecutive ones in the n-bit code words and k indicates a maximum number of zeros between consecutive ones in the n-bit code words.

37. The method of claim 36, wherein  $m/n$  is greater than  $\frac{1}{2}$ , and  $d=2$ .

38. The method of claim 37, wherein  $r1+r2+r3$  equals 9.

39. The method of claim 36, wherein the n-bit code words of the first type end in "00", the n-bit code words of the second type end in "10" and the n-bit code words of the third type end in "01".

40. The method of claim 39, wherein the n-bit code words in a coding state of the first kind start with "00", the n-bit code words in a coding state of the second kind start with "00", "01" or "10", and the n-bit code words in a coding state of the third kind start with "00" or "01".

41. The method of claim 32, further comprising:

receiving a modulated signal; and

demodulating the modulated signal into at least the n-bit code words.

42. The method of claim 32, further comprising:

reproducing a modulated signal from a recording medium; and

demodulating the modulated signal into at least the n-bit code words.

43. A decoding device, comprising:

a converter receiving n-bit code words, where n is an integer, and converting the n-bit code words into m-bit information words, where m is an integer less than n, the n-bit code words being divided into a first type, a second type and a third type and into coding states of a first kind, a second kind and a third kind such that an m-bit information word is converted into an n-bit code word of the first, second or third kind if the previous m-bit information word was converted into an n-bit code word of the first type and is converted into an n-bit code word of the first or third kind, if the previous m-bit information word was converted into an n-bit code word of the second type and is converted into an n-bit code word of the first kind, if the previous m-bit information word was converted into an n-bit code word of the third type, wherein sets of code words

belonging to the different coding states do not contain any code words in common.

44. The decoding device of claim 43, wherein the n-bit code words are divided into r1 coding states of the first kind, r2 coding states of the second kind and r3 coding states of the third kind, where r1, r2 and r3 are integers greater than or equal to 1, and each of the r1, r2 and r3 coding states have n-bit code words different from the n-bit code words in the other r1, r2 and r3 coding states.

45. The decoding device of claim 44, wherein the converter determines to which of the r1, r2 and r3 coding states a next n-bit code word belongs, and converts a current n-bit code word into an m-bit information word based on the determined coding state.

46. The decoding device of claim 45, wherein at least one of the r1, r2 and r3 coding states includes more than one of a same n-bit code word, the same n-bit code word maps to more than one of the m-bit information words, and each same n-bit code word has a different state direction associated therewith, each state direction indicating a next one of the r1, r2 and r3 coding states from which to obtain the next n-bit code word when converting the m-bit information words into the n-bit code words.

47. The decoding device of claim 46, wherein the n-bit code words satisfy a dk-constraint, where d indicates a minimum number of zeros between consecutive ones in the n-bit code words and k indicates a maximum number of zeros between consecutive ones in the n-bit code words.

48. The decoding device of claim 47, wherein  $m/n$  is greater than  $\frac{1}{2}$ , and  $d=2$ .

49. The decoding device of claim 48, wherein  $r1+r2+r3$  equals 9.

50. The decoding device of claim 49, wherein the n-bit code words of the first type end in "00", the n-bit code words of the second type end in "10" and the n-bit code words of the third type end in "01".

51. The decoding device of claim 50, wherein the n-bit code words in a coding state of the first kind start with "00", the n-bit code words in a coding state of the second kind start with "00", "01" or "10", and the n-bit code words in a coding state of the third kind start with "00" or "01".

52. The decoding device of claim 43, further comprising: a demodulator receiving a modulated signal and demodulating the modulated signal into at least the n-bit code words.

53. The decoding device of claim 43, further comprising: a reproducing device reproducing a modulated signal from a recording medium, and demodulating the modulated signal into at least the n-bit code words.

\* \* \* \* \*